DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GOR/ENS/98M-15

A NETWORK DISRUPTION MODELING TOOL

Capt James A. Leinart

AFIT/GOR/ENS/98M-15

Approved for public release; distribution unlimited

AFIT/GOR/ENS/98M-15

A NETWORK DISRUPTION MODELING TOOL

THESIS

Presented to the Faculty of the Graduate School of Engineering of the

Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Operations Research

James A. Leinart, B.S.

Captain, USAF

MARCH 1998

Approved for public release; distribution unlimited

THESIS APPROVAL

Student: James A. Leinart, Captain, USAF Class: GOR-98M

Title: A Network Disruption Modeling Tool

Defense Date: 17 March 1998

Committee:	Name/Title/Department	Signature
Advisor	Richard F. Deckro Professor of Operations Research Department of Operational Sciences	
Reader	Jack A. Jackson Jr., Lieutenant Colonel, USAF Assistant Professor of Operations Research Department of Operational Sciences	
Reader	Jack M. Kloeber Jr., Lieutenant Colonel, USA Assistant Professor of Operations Research Department of Operational Sciences	

DISCLAIMER

The views expressed in this article are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government. In addition, the weightings and single dimensional value functions used in the model were for academic purposes only and as such do not represent the opinions of the National Air Intelligence Agency.

DISCLAIMER

Acknowledgments

First and foremost, I want to thank God who always causes me to triumph in Christ Jesus and whose plans of peace toward me have brought me to an expected end. Next, I wish to thank my wonderful, incredibly supportive wife, Kristine. Her listening ear and wise words brought me rest and encouragement throughout the thesis process. I also want to thank my family and friends, who have encouraged me along my AFIT path.

Within the AFIT environment, Dr. Deckro deserves the most thanks from me. He stretched my mind, causing me to think out of the box, and opened my eyes to research and analysis approaches and thoughts, that I am unsure I would ever have learned in all of my upcoming analyst career if it were not for him. Not only did he provide me with outstanding academic guidance, he also encouraged me when I was overwhelmed with the thesis effort. Throughout this process, he has become a mentor and friend to me.

LtCol Jackson and LTC Kloeber deserve many thanks for their encouragement, patience and insight into the problem I was tackling. LtCol Jackson provided me with a "real world" perspective of my problem, while LTC Kloeber was a master at teaching me many valuable analytical concepts and giving me helpful comments in his role as a reader.

I also wish to thank Mr. Keith Anthony and Mr. Jim Homer at NAIC for their guidance, listening ears, and timely data provision. Additionally, Doug Young and friends at the NSA also must be thanked for providing software and ideas.

Lastly, my classmates and other teachers must be thanked for encouraging me, teaching me and rallying around me as the thesis process drew to a close.

Table of Contents

List of Figures

List of Tables

Abstract

Given that network disruption has been identified as a military objective and C2-attack has been identified as the mechanism to accomplish this objective, a target set must be acquired and priorities assigned to each target set. In order to move beyond subjective target prioritization, development and implementation of a methodology and model, that quantitatively measures the value of each target in achieving the objective, is needed in prioritizing targets.

This thesis effort uses a vertex cut-set algorithm on a transformation of the graph representing a notional network of interest. The cut-sets generated represent potential target sets. Value-focused thinking and decision analysis techniques are used to rank the target sets according to decision maker preferences and the overall objective of achieving network disruption. The ranked list of potential targets can be narrowed down to an amount palatable to the decision maker, C2W planner, or some other user, so that further analysis may be conducted.

The above methodology is incorporated in a Visual Basic/Excel Spreadsheet environment and allows for user-friendly, yet powerful analysis.

A NETWORK DISRUPTION MODELING TOOL

I. Introduction

Background

"Automated information systems and networks provide the predominant source from which the warfighter generates, receives, shares, and utilizes information" [Joint Pub 6-0, 1995: preface]. These words of General John M. Shalikashvili, retired Chairman of the Joint Chiefs of Staff (JCS), underscore the critical role that information networks play in the operations of all the Services. This point is further emphasized in General Shalikashvili's statement, "Command, control, communications, and computer (C4) networks and systems provide the means to synchronize joint forces" [Joint Pub 6-0, 1995: preface]. The above quotations highlight the heavy dependence of the United States military on C4 networks and systems, and wherever dependence exists, vulnerabilities and opportunities for exploitation must be a serious consideration. These statements are not only true for the United States, but are also true for other technologically advanced countries.

The Centre for Infrastructural Warfare Studies (CIWARS), a threat analysis think-tank, publishes an Infrastructural Warfare Situation Report (ISR), which categorizes countries into three phases. These phases characterize the industrial and technological maturity of each country, with phase three used to describe countries that are, "Most dependent on industrial infrastructure. Includes well developed telecommunications,

electric systems, high rate of personal computers, high percent of skilled workers, …" [Journal for Infrastructural Warfare, 1997: 1]. The United States, in order to more effectively defend its interests, must be able to exploit the vulnerabilities of the C4 systems and networks upon which our adversaries depend.

Information Operations (IO) are the means by which the United States can exploit enemy C4 system and network vulnerabilities. The first draft of Joint Publication 3-13 states IO are, "Actions taken to affect adversary information, and information systems, while defending one's own information and information systems" [Joint Pub 3-13, 1997: GL-11]. A subset of IO, conducted during a crisis or war in order to achieve specific objectives over a specific adversary or adversaries, is information warfare (IW) [Joint Pub 3-13, 1997:GL-11]. In turn, IW also contains a subset - command and control warfare (C2W) [Joint Pub 3-13, 1997:GL-6]. Joint Publication 3-13.1 states, "Command and control Warfare (C2W) is an application of IW in military operations and employs various techniques and technologies to attack or protect a specific target set-command and control (C2)" [Joint Pub 3-13.1, 1996: v].

C2W has both an offensive and a defensive face, C2-attack and C2-protect [Joint Pub 3-13.1, 1996: I-4]. This thesis focuses on the offensive element of C2W. C2-attack is used to, "Prevent effective C2 of adversary forces by denying information to, influencing, degrading, or destroying the adversary C2 system" [Joint Pub 3-13.1, 1996: I-4]. Since computers and communication are tools for use in commanding and controlling forces and their operations, a C2-attack operation may target anyone or all four parts of a C4 system. An effective way to attack the above command and control

target set is to C2-attack the physical and logical components of an enemy's C4 system. The major components of a C4 system include: terminal devices such as telephones and computer; transmission media used to connect the terminal devices; switches which route traffic through a transmission media network; and control, broken into two parts, network and nodal. These components provide access to networks [Joint Pub 6-0, 1995: viii, ix].

> Networks are formed when terminal devices and transmission media are interconnected with switching equipment to ensure that information (voice, imagery, data, or message) is transported to appropriate locations. The networks that result from open systems architectures are called information grids. They allow warriors to gain access to, process, and transport information in near real time to anyone else on the network. [Joint Pub 6-0, 1995: ix]

Through proper C2-attack of the major system components, network disruption should result, thereby limiting or nullifying the intended functionality of the enemy's networks. Joint Publication 3-13.1 echoes this sentiment, as it describes a mental effect as well as a physical effect, "Effective C2W operations influence, disrupt or delay the adversary's decision cycle" [Joint Pub 3-13.1, 1996: I-6]. Throughout this thesis, the definition of network disruption consists of two key parts: (1) severance or hindrance of information flow, (2) and the nodes - between which flow is to be stopped or hindered.

In order for the C2-attack of the major system components to be effective, proper targeting must occur. Air Force Instruction (AFI) 14-207 provides guidance on air targeting [AFI 14-207, 1993: 1]. This instruction outlines six phases of the targeting process as seen in Figure 1-1. The second phase of the targeting process is target development and is the phase of interest for this thesis [AFI 14-207, 1993: 3-4]. Target

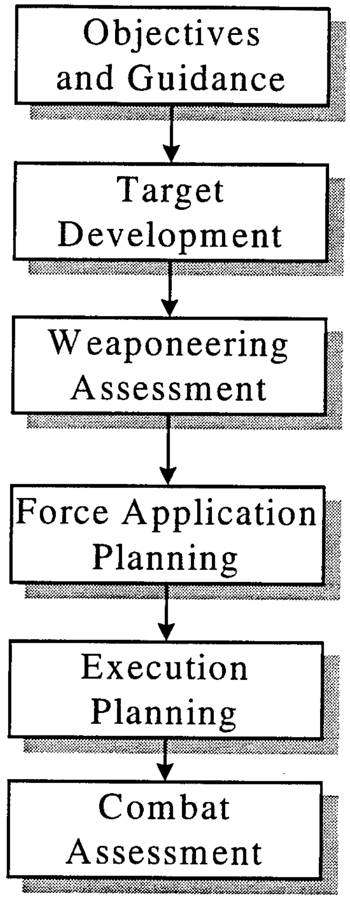

Figure 1-1. Targeting Process

development includes evaluating potential targets and their components to identify vulnerabilities, and compiling a prioritized list of potential targets [AFI 14-207, 1993: 4]. Important factors to consider throughout the targeting process are the *benefits* and *costs* resulting from attacking the targets. Analysis of target characteristics should provide insight not only into target features, but should also aid in determining the tradeoff between attack benefits and costs associated with the target.

Major system components are critical for the proper operation of a C4 network. These components can be described by their characteristics. Targeting and attacking

these components through a C2-attack operation can help achieve the overall objective of network disruption; however, a target analyst must be aware of the potential *benefits* and *costs* related to the targets.

Statement of the Problem

Once network disruption has been identified as a military objective and C2-attack has been identified as the mechanism to accomplish this objective, a target set must be acquired and priorities assigned to each set. To assist in the process of target prioritization, development and implementation of a methodology that quantitatively measures the value of each target set is needed.

Research Approach

The first step in solving this problem involves characterizing the network as a graph, and analyzing the major system components, which form a network, to determine their key characteristics. A cut-set generation algorithm is then employed to generate potential target sets. Decision analysis (DA) and value focused thinking (VFT) are used to develop a "first cut" value hierarchy of the benefits and costs associated with attacking each system component, relative to the overall objective of network disruption.

Once the hierarchy has been developed, components, which are potential targets, can be evaluated by measuring their "costs" and "benefits" via their characteristics. The application of a function to the values of the targets in a target set gives the target set's value. The target sets are then ranked by value with this ranking creating a prioritized list of candidate target sets.

The above steps are accomplished via a Visual Basic (VB) package designed to interface with an Excel spreadsheet, where the user inputs the C4 network components, the components' attributes, and hierarchy data.

Scope/Limitations

The network of focus for this research is a voice telecommunications network; although, the methodology used applies to other types of networks. The telecommunications network will be decomposed into four major parts:

1) The ground telephone system
2) The cellular phone network
3) The radio telephone system
4) The satellite telecommunications network

Each of these parts will be broken down into components comparable to those in Joint Publication 6-0, with the exception of the control element.

The components of the system form the nodes and links of the network and thus become the candidate targets for C2-attack. It is assumed that effective control is lost once communication is lost. The nodes in this proof-of-concept study will be considered non-mobile since the model will portray a snapshot in time representation of the network. This is not unreasonable given good battlespace preparation intelligence coupled with the fact that the attack is assumed to have an H-hour, with satellites and mobile telephone users in known (or approximately known) locations. Of course, complete knowledge of an opposition's system may not be possible. The model's analysis is based on a known network system.

For the purposes of this research, network disruption does not involve corrupting the actual information, but instead is concerned with stopping or hindering the flow of information. It is assumed that enemy C4 systems and networks resemble those of the US military and have similar components; while this is certainly valid in many cases, there may be some countries that have a different C4 system paradigm. However, as the US military C4 systems and networks are among the best in the world, the evaluation should provide very useful results, which can be applied to networks of varying quality. The results of the analysis will focus on the individual components, along with the associated benefits and costs from attacking the components. A more detailed analysis is necessary to capture both synergistic and substitution effects. Lastly, the values will be fairly general and only preferences, varying between decision makers.

Thesis Overview

Chapter I has focused on background information relating to the problem of prioritizing targets to accomplish network disruption. Chapter II reviews network component descriptions, introduces graph theory, outlines network cut-set determination, and outlines decision analysis methods. Chapter III provides a cohesive methodology which can be used to solve the different phases of the research problem. Chapter IV analyzes a notional example and the C2-attack results, and Chapter V provides the overall conclusions, as well as recommendations for future related research.

II. Literature Review

Introduction

This chapter provides the relevant terms, concepts, and exposure in the literature to attack this problem area. Independent sections discuss C4 and telecommunication networks, networks and graph theory, and decision analysis and value focused thinking.

C4 and Telecommunications Networks

Joint Publication 6-0 is the key document for the series of military publications concerning C4 systems, and provides guidance and doctrine in the realm of C4 systems support [Joint Pub 6-0, 1995: i]. It states that C4 networks are formed by interconnection of the following four major C4 systems components: terminal devices such as telephones and computers; transmission media used to connect the terminal devices; switches which route traffic through a transmission media network; and control, broken into two parts, network and nodal [Joint Pub 6-0, 1995: viii]. Joint Publication 6-0 goes on to define each of the four components.

Terminal devices are essentially devices which convert information comprehensible to the warfighter into a format for electronic transmission, or vice-versa [Joint Pub 6-0, 1995: viii]. Transmission media are the conduits through which electronic information travels between the terminal devices. Joint Publication 6-0 states, "There are three basic electronic transmission media: radio (including space based systems), metallic wire, and fiber-optic cable" [Joint Pub 6-0, 1995: viii]. Switches are simply devices used for routing information through the transmission media network.

C4 systems control is equivalent to a management mechanism for C4 systems, and can be divided into two levels. The first level is nodal control, which is concerned with managing local C4 systems. The second level, network control, entails a wider focus than nodal control and is principally concerned with the management and configuration between facilities under nodal control [Joint Pub 6-0, 1995: ix]. These four components, terminal devices, transmission media, switches, and control, form the logical and physical links and nodes of the overall C4 network.

In this effort, voice telecommunication networks serve as the C4 networks of interest. The network is decomposed into four major parts:

1) The ground telephone system
2) The cellular phone network
3) The radio telephone system
4) The satellite telecommunications network

Each of these parts is broken down into components, which are either a subset of or analogous to the four major systems components mentioned in Joint Publication 6-0. Additionally, it is assumed that the components making up these parts will be the nodes and links of any telecommunications network. Each of these components have characteristics that identify their role and interaction in accomplishing information flow in the network.

Saadawi, Ammar, and Hakeem [1994] state and define many of the components of a telecommunication system. Likewise, Miller and Ahamed [1988] discuss some of the same components. Each of the four main parts of the telecommunications network and their components will be discussed in turn, and their relationship to the systems components mentioned in Joint Publication 6-0 will be stated. With such a relationship,

the eligibility and viability of these components as C2-attack targets, for this effort's network, will be clear.

The Ground Telephone System. Saadawi, Ammar, and Hakeem [1994] discuss two components of telecommunications networks, which can be applied to ground telephone networks. The first component is the *transmission facility*. A transmission facility can be broken into two parts: the *local loop* and the *trunk lines*. The local loop connects equipment, such as telephones, with the telephone company's switching office, also know as the central office or local exchange. Many of the telephone local loops are wire-pair cables, although a large percentage of newer installations utilize fiber-optic cables. Trunk lines, also referred to as circuits, connect two switching systems. Trunk lines carry traffic generated by a large amount of customers, whereas loops are dedicated to individual customers. Wire pairs, coaxial cables, microwave radio, satellites, and fiber optics are types of transmission media employed in the use of trunk lines [Saadawi, Ammar, and Hakeem, 1994: 22]. The transmission facilities are analogous to C4 system paths, referred to in Joint Publication 6-0, over which information travels. The transmission media employed in the use of trunk lines are also listed in Joint Publication 6-0.

The second component is the *switching system*, whose function is to connect circuits and route traffic through the network. Switches essentially remove the need for a direct line between every piece of telecommunications equipment in the network. For a telephone network, there are two groups into which switching systems can be placed; *local* and *tandem* switching systems. The local switching systems, termed *central office* (CO) switches, are used to connect customer loops directly to other customer loops or

customer loops to trunks. Tandem switches connect either trunks to trunks or CO switches to CO switches. A tandem switch that serves the long distance network is a *toll* switch [Saadawi, Ammar, and Hakeem, 1994: 22]. These switching systems are comparable to those mentioned in Joint Publication 6-0.

Not all features and functions of a ground telephone network fit exactly into the framework of C4 systems mentioned in Joint Publication 6-0. Martin introduces one component, the repeater, which is a device used to restore signals distorted because of attenuation [Martin, 1976: 645]. It is not a switching system, and even though a repeater is used by transmission facilities, it is not such a facility. Despite this lack of conformity to the C4 systems framework in Joint Publication 6-0, the repeater is an important component and is taken into account in this research.

<u>The Cellular Phone Network.</u> Mobile (cellular) telephone systems are circuit-switched and use radio frequency transmission [Saadawi, Ammar, and Hakeem, 1994: 26]. Circuit switching is the most common method for switching in a telephone network. When communication is desired between two customers, via their telecommunications equipment, this technique establishes a dedicated path. This path, which is a connected sequence of links, is formed by capturing channels, before any transmission takes place, and is maintained for the duration of the communication [Miller and Ahamed, 1988: 116-117; Saadawi, Ammar, and Hakeem, 1994: 10].

A mobile telephone network can be divided into three main parts: the user, along with the mobile phone; a cell site, to capture the user's signal and connect it with the terrestrial telephone network; and a *mobile telephone switching office*, MTSO, wired to

and controlling all radio towers and routing calls into the public telephone network [Saadawi, Ammar, and Hakeem, 1994: 26-27]. The user's mobile phone falls in the category terminal devices, outlined in Joint Publication 6-0. Within the publication, switches were also mentioned; the MTSO is such a device.

The cell site contains a radio tower and can be considered similar to a repeater. As such, it does not fit neatly into any component category of Joint Publication 6-0. The radio transmissions between users and the cell site radio tower are links of the mobile telephone network, and are a type of transmission media named in Joint Publication 6-0. Additionally, the wires connecting the cell sites and the MTSO are network links.

The Radio Telephone System. This system is essentially composed of radio towers. For the illustrative example in this research, microwave radio systems will be the system type of interest. These systems are generally constructed with radio towers spaced about every 30 miles apart. These towers are the relay points in an end to end transmission and contain repeaters to amplify the signal [Martin, 1976: 166-167]. The only component of microwave telephone systems correlated to those mentioned in Joint Publication 6-0 is the microwave transmission media linking the radio system together.

The Satellite Telecommunications Network. In essence, the satellite telecommunications network is a microwave radio system with only one repeater, the satellite transponder in orbit. Earth stations transmit, or uplink, information to the satellite, and the satellite transmits, or downlinks, the information back down to earth stations [Saadawi, Ammar, and Hakeem, 1994: 24-25]. For this effort, only one geosynchronous satellite will be considered. It does not fall into any of the four main

components of Joint Publication 6-0. Earth stations will be considered as a subset of terminal devices discussed in Joint Publication 6-0, and the microwave radio transmissions, between the earth stations and the satellite, a subset of the aforementioned transmission media. The satellite and earth stations are characterized as network nodes, while the microwave radio transmission channels are characterized as links.

Telephone Network Representation. Table 2-1 shows the components of the four parts of the telecommunications network. The components' representations as either nodes or links, for the network of this effort, are also shown.

Table 2-1. Component Representations

Representations	Ground	Cellular	Radio	Satellite
Nodes	Switches	Cell sites	tower	satellite
	Repeaters	MTSO		earth station
Links	local loops	radio channels	microwave channels	microwave channels
	trunk lines	MTSO to cell wiring		

Table 2-2 identifies whether a network component used in this effort is comparable to a component identified in Joint Publication 6-0.

Table 2-2. Component Comparison to Joint Publication 6-0 Components

	Representation	Ground	Cellular	Radio	Satellite
Inside JP 6-0	Terminal Devices				earth station
	Transmission Media	local loops trunk lines	radio channels wiring	microwave channels	microwave channels
	Switches	switching systems	MTSO		
Outside JP 6-0		repeaters	cell site	tower	satellite

All these components, along with their features, physically represent the network; however, a theoretical representation of the components and the network is available through the use of graph theory.

Networks and Graph Theory

Networks of interest can often be mathematically modeled by graphs [Frank and Frisch, 1971: 1]. A graph, G, is composed of a set of vertices, V, and a set of edges, E and may be symbolized as G(V, E). V is comprised of n vertices each labeled, v_i, where i = 1, 2, 3, ..., n, and E is comprised of m edges each labeled, e_j, j = 1, 2, 3..., m. In this study, the vertices of G correspond to the nodes of the communication network and the edges of G correspond to the links of the network.

Graphs can be directed or undirected. If they are directed then their edges are oriented from one vertex to another vertex, and flow can only occur in the direction of the orientation. However, if the graph is undirected, there is no "orientation" associated with

an edge, and flow may occur in either direction simultaneously. For this research project, it is assumed that the graph is undirected.

Associated with each edge and vertex in the graph are certain attributes or characteristics of interest. The attributes provide the means to numerically state relevant parameters of the network [Frank and Frisch, 1971: 1; Evans and Minieka, 1992: 5]. One example, of many possible, is edge capacity, c. The capacity of an edge, $c(e_j)$, is the maximum amount of flow that the corresponding link can accommodate. For the purposes of this thesis, capacity is measured in the number of voice channels available.

Other useful graph concepts are chains, connected graphs, subgraphs, proper-subgraphs, components, vertex cut-sets, edge cut-sets and mixed cut-sets. Suppose a given graph G(V,E) has at least three vertices and two undirected edges, and assume vertices, v_1 and v_2, are connected by edge e_1, and vertices, v_2 and v_3, are connected by edge e_2. The sequence of edges and vertices v_1, e_1, v_2, e_2, v_3 is a chain from vertex one to vertex three [Bazaraa, Jarvis, and Sherali, 1977: 422; Ford and Fulkerson, 1962: 3].

Bazaraa, Jarvis, and Sherali state that a *connected graph* is a graph which has a chain from every vertex to every other vertex in the graph. They go on to define a subgraph, $G'(V',E')$ of G(V,E), as a graph where both $V' \subseteq V$ and $E' \subseteq E$. It is also assumed that if $e_i \in E'$ and connects v_i and v_{i+1}, then both v_i and v_{i+1} are elements of V'. Additionally, if $G' \neq G$, then G' is a proper-subgraph of G. Encapsulating these previous concepts is a component. A graph component is a connected subgraph which is not a proper-subgraph of another connected subgraph [Bazaraa, Jarvis, and Sherali, 1977: 423].

A *vertex cut-set of a graph*, with undirected edges only, is a minimal set of vertices, whose removal from a graph divides the graph into more than one component. There is no proper subset of a vertex cut-set since the cut-set is composed of a minimal set of vertices. It is assumed that the edges incident to the vertex are also removed, when a vertex is removed from a graph [Frank and Frisch, 1971: 18-19].

An *edge cut-set of a graph*, with undirected edges only, is a set of a minimal number of edges, whose removal from a graph divides the graph into more than one component, and an edge cut-set also has no proper subset [Frank and Frisch, 1971: 17-18].

A *mixed cut-set of a graph* is a "combination" of vertex and edge cut-sets. It is the minimal set of both vertices and edges, whose removal breaks all chains between two specified vertices. For example, if a graph G contains both v_s and v_t, then a s-t mixed cut-set is the minimal set of edges and vertices, other than v_s and v_t, whose removal breaks all chains between v_s and v_t [Frank and Frisch, 1971: 302].

Cut-set Generation. Various algorithms for generation of vertex and edge cut-sets have been developed throughout the years. Several algorithms use a combination of boolean algebra and either the minimal paths or basic minimal paths of a graph to generate all the edge or vertex cut-sets of a graph. Given a graph with a source vertex, s, and a sink vertex, t, a minimal path,

> ... is a set of edges such that (i) it is possible to traverse from s to t along these edges and (ii) at every vertex in P other than s and t only two edges are incident. In the vertex representation, a minimal path constitutes an ordered sequence of vertices, whereas in edge representation it constitutes an ordered sequence of edges. [Prasad, Sankar and Rao, 1992: 1293]

"A basic minimal path is a minimal path such that if two vertices are not adjacent in a path, they are not adjacent in the graph also" [Prasad, Sankar and Rao, 1992: 1293]. The cut-sets are minimal groupings of these paths and the groupings are determined by boolean algebra [Biegel, 1977: 39].

There exists another, completely different approach for generating vertex cut-sets of an undirected graph. Suppose an undirected graph contains, among other vertices, two vertices, s and t, and it is desired to find all the vertex cut-sets which separate these two vertices. The approach involves generating different subgraphs containing vertex s. As each subgraph is generated, vertex sets which separate the subgraph, containing s, from vertex t and any subgraph to which it may belong are examined according to certain criteria. The vertex sets which meet the criteria are vertex cut-sets.

Patvardhan, Prasad, and Pyara developed an algorithm for generating all vertex cut-sets of an undirected graph using the above approach [Patvardhan, Prasad, and Pyara, 1995]. Given that an undirected graph can be transformed so that the original edges are represented as vertices, application of the algorithm to the transformed graph yields all possible cut-sets of the original graph. Determination of the cut-set type, vertex, edge or mixed, is easily derived by matching the generated cut-set vertices with the original graph's edges and vertices.

The algorithm developed by Patvardhan, Prasad, and Pyara was chosen, for use in this effort, for two primary reasons. First, it was applicable, without any modification, to this study's communication network representation, *viz.*, an undirected graph.

Additionally, the algorithm's time and space complexity were well established and provided known bounds for the problem of interest. In particular, the algorithm has a space complexity of $O(n^2)$ and a time complexity of O(mn) per vertex cut-set, where m is the number of edges in the graph and n the number of edges approach [Patvardhan, Prasad, and Pyara, 1995].

Decision Analysis and Value Focused Thinking

In the course of making hard decisions, a decision maker can easily stray from keeping clearly in focus the ideals which really matter to him or her. Often politics, pressures, and other competing interests blur the choices that a decision maker has to make, and the challenge to the decision maker is to not veer from what is important in the decision situation at hand. Decision analysis and value focused thinking are two ways for a decision maker to keep in mind what is important; through quantifiably measuring the importance, value, of a decision alternative.

The first tool, decision analysis, mentioned above is well described in the words of Robert T. Clemen,

> … the objective of decision analysis is to help a decision maker think hard about the specific problem at hand, including the overall structure of the problem as well as his or her preferences and beliefs. Decision analysis provides both an overall paradigm and a set of tools with which a decision maker can construct and analyze a model of a decision situation" [Clemen, 1996: xix].

Clemen developed a flowchart to guide the decision maker through the decision analysis process on the way to a successful achievement of the objective of DA [Clemen, 1996: 5-8]. The flowchart is shown in Figure 2-1. Kirkwood, another leader in

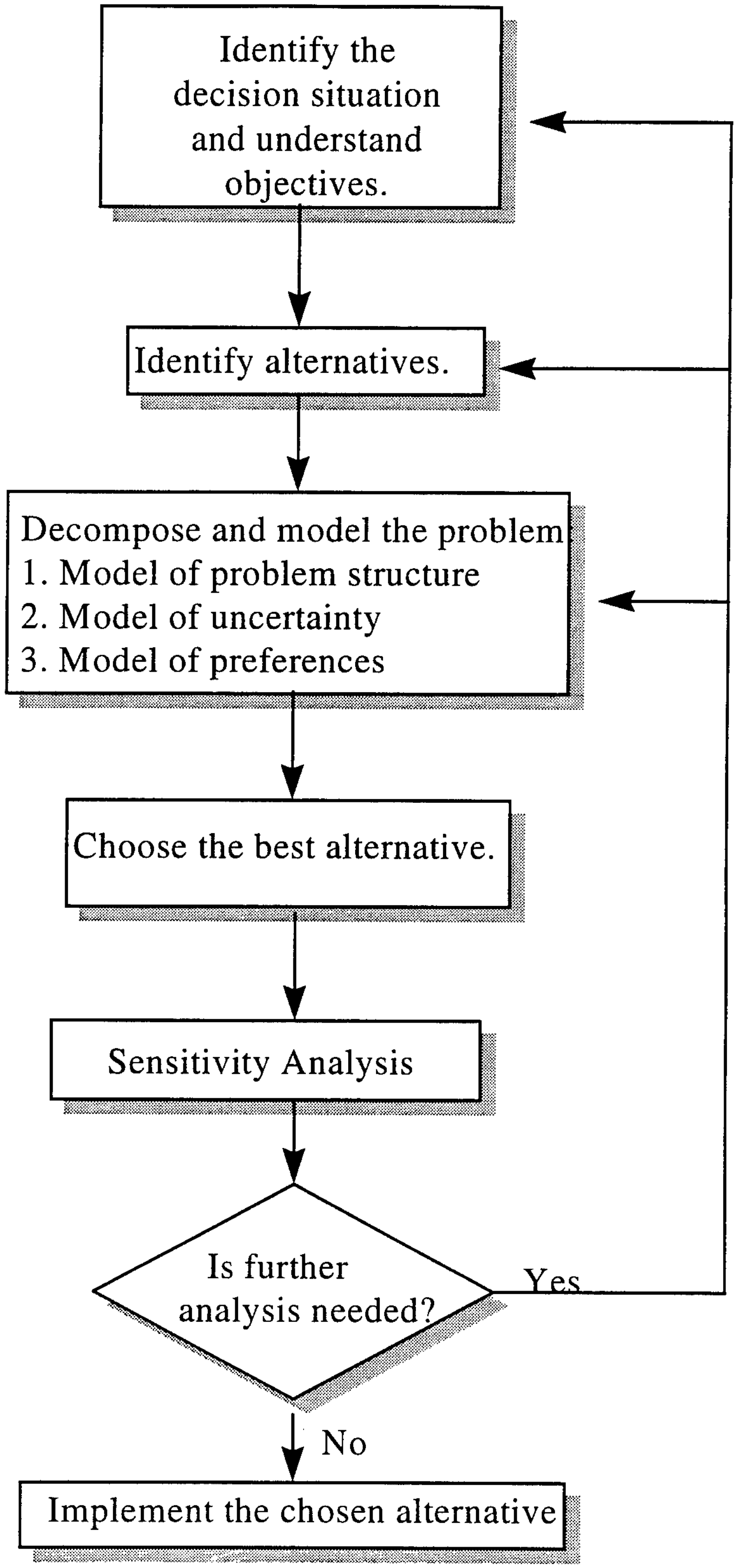

Figure 2-1. Decision Analysis Flowchart

DA, lists five specific steps to help a decision maker achieve the objective of decision analysis. These steps are similar to those presented by Clemen:

1) Specification of objectives and scales used for achievement measuring with respect to the objectives
2) Development of alternatives that possibly might achieve the objectives
3) Determination of how well each objective is achieved by each alternative
4) Analysis of tradeoffs between the objectives
5) Selection of the alternative that, overall, best achieves the objectives, and accounts for uncertainty [Kirkwood, 1997: 3]

Deciding what is important is an issue that needs to be addressed in dealing with objectives for a decision analysis [Kirkwood, 1997: 11]. Clemen and Keeney call the things that are important values, and Kirkwood names them evaluation considerations [Clemen, 1996: 19; Keeney, 1992: viii; Kirkwood, 1997: 11]. Keeney states that many decisions focus on alternatives, but since values are what really matter in a decision, values are where the focus should be. Keeney's school of thought is known as value focused thinking [Keeney, 1992: viii].

Values can sometimes be decomposed into a hierarchical format, often with several layers. These layers of evaluation considerations continue to a depth dependent upon the decision at hand and decision maker preferences. Each layer describes, in more detail, the above layer. A measure of effectiveness, or metric, is a measuring scale for determining to what degree an objective has been attained. The numerical rating that a

decision alternative obtains with respect to a particular evaluation measure is called the score of an alternative [Kirkwood, 1997: 11-13].

In order for the value hierarchies to be functional, certain properties are desired. The first is completeness, which means that, "... the evaluation considerations at each layer (tier) in the hierarchy, taken together as a group, must adequately cover all concerns necessary to evaluate the overall objective of the decision." [Kirkwood, 1997: 16] In addition to the fact that all evaluation concerns must be adequately covered by the lowest layer evaluation considerations, the evaluation measures for the lowest layer evaluation concerns must adequately measure the degree of attainment of their related objectives.

The second desired property is nonredundancy. This property means that within a layer of the hierarchy, no two evaluation considerations should overlap. This property ensures that evaluation considerations are not counted twice. The third property, independence, goes one step beyond nonredundancy. The essence of this property is that not only must the evaluation considerations not overlap, but also must be independent [Kirkwood, 1997: 16-18].

Value functions are used to convert the information from value hierarchies into something meaningful to the decision maker. In DA, a multicriteria value function is used when there are multiple objectives that conflict and the outcome of each alternative is known with certainty. The various evaluation measures are combined, through the use of the multicriteria value function, into a single measure of the overall value of each alternative. For example, suppose there exist several alternatives in a decision, which has been decomposed into its evaluation considerations. At their lowest layers of the

hierarchy, assume each evaluation consideration has a different evaluation measure, x_i. A function is then needed for each evaluation measure which converts the different evaluation measures into an equivalent, common rating. This rating is value. The converting function is called a single dimensional value function and labeled as v_i, where the subscript denotes the application of the value function to ith evaluation measure, x_i. Weights are also assigned to each evaluation measure to account for differences in variation ranges for the evaluation measures and differing degrees of importance attached to the variation ranges. Each weight is labeled as w_i, and the sum of the weights equals one. The products of each evaluation measure's weight and the evaluation measure's single dimensional value function are summed to form the multiobjective value function, v. The mathematical representation of the multiobjective value function is $v(x) = \sum_{i=1}^{n} w_i v_i(x_i)$, with the assumption that the hierarchy contains n evaluation measures. This multiobjective value function can then be used to rank the alternatives [Kirkwood, 1997: 53, 59-61].

Davis employed DA and VFT to help decision makers balance conflicting objectives which exist when planning network expansion [Davis, 1997: 1-5]. This thesis is concerned with a problem of an opposite nature to Davis' work. Instead of expanding a network efficiently, selection of targets for disruption and degradation of a network is accomplished through decision analysis techniques.

Summary

This chapter presented key concepts and definitions in the literature related to this thesis research. The highlights are: 1) The telecommunications network can be divided into four main parts, which can be subdivided into components represented as network nodes and links; 2) graph and network theory provide methods to gain insight and useful information from such a representation; and 3) value focused thinking is used in determining the values of node and link sets. Finally, the network is scored based on these values. Chapter III discusses in detail the relevant and appropriate application of these highlights.

III. Methodology

The methodology for evaluating the telecommunications network components and choosing which to C2-attack is developed in this chapter. The methodology is a combination of two problem-solving tactics: network analysis techniques, used for identification of target sets, and value-focused thinking and multiobjective decision analysis, which are used to measure the value of the network components. Figure 3-1 shows the stages of the methodology, along with the tactic employed, and the output produced at each stage. A preliminary description of the situation is given before the explanation of the methodological tactics.

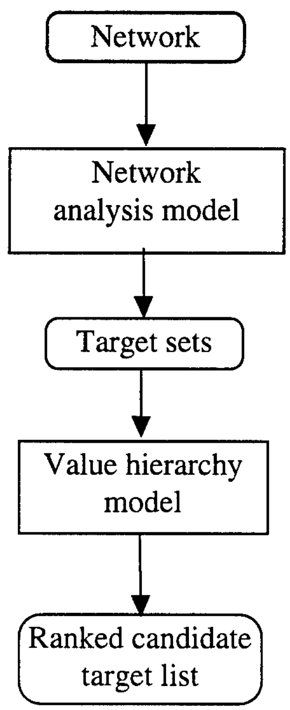

Figure 3-1. Flow chart of Methodology

Background

The telecommunications network of interest is used by an adversary for the purposes of command and control. Disruption of communication between specific users of the network is required in order to diminish the enemy's command and control capabilities. The specified users are located at geographically separated nodes within the network, and the network has components, listed in Table 2-1, pre-identified as potential targets. All possible target combinations, whose removal results in separation of the specified users, are the target sets, and these sets can be used in the development of a prioritized candidate target set list can be developed.

The network itself is notional, fabricated from information in open sources [Alliance for Telecommunications Industry Solutions. T1A1.2 Working Group on Network Survivability Performance, 1993; Bunger, 1998; Couch, 1995; Flood, 1975; http://www.crtc.gc.ca/eng/telecom/decision/1996/d9613_0.txt, 1996; Intelsat, 1998; Martin, 1976; Miller and Ahamed, 1988; Saadawi, Ammar, and Hakeem, 1994] (See Figure 3-2).

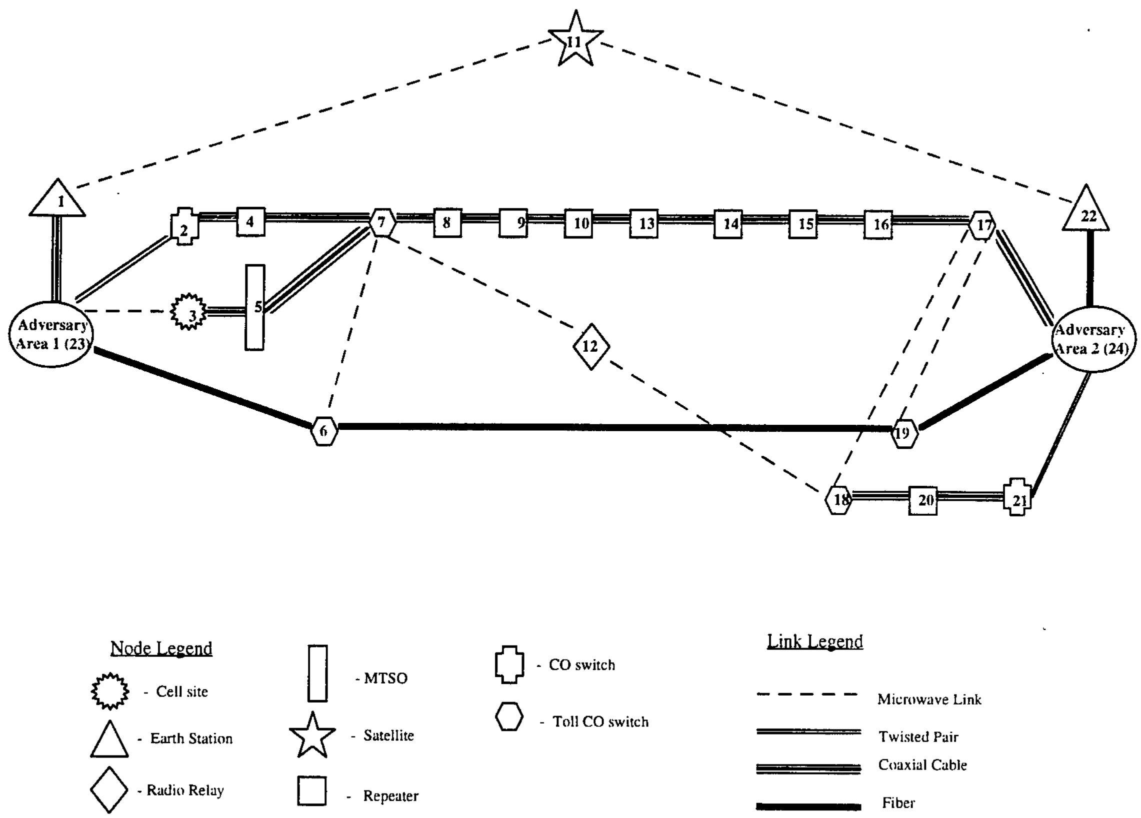

Figure 3-2. Notional Telecommunications Network

The Network Analysis Model.

The network analysis model accomplishes target set generation by taking as input the vertex adjacency representation of the network, transforming the graph represented by the adjacency list, and then applying a cut-set enumeration algorithm to the graph. A detailed discussion of this analysis process follows.

Graph Representation of the Network. In the telecommunications network shown in Figure 3-2, the adversary users are located in the nodes, *Adversary Area 1* and *Adversary Area 2*, and these nodes represent a group of many users in close proximity to each other.

In the network analysis model, the network is represented as an undirected graph, where the network nodes and links are represented as the graph's vertices and edges, respectively. Since the graph is undirected, there is neither a source vertex nor sink vertex, *per se*; however, for identification of the nodes, between which communication is to be disrupted, the node *Adversary Area 1* is labeled as either the source or sink vertex, and the node *Adversary Area 2* is labeled oppositely.

Graph Transformation. In the analysis process, the graph representation of the network, is transformed into an equivalent, yet alternative graph. This is done by replacing every edge in the original graph with a vertex and two edges, where the two edges connect the newly created vertex to the original vertices joined by the replaced edge. Consideration of this alternative graph's vertices, reveals that its vertices correspond not only to the network's nodes, but also to the network's links, and the links of the alternative graph are merely placeholders for showing how the network's nodes and links are related. This transformed graph is represented as a vertex adjacency list, which is the appropriate input format for the cut-set enumeration algorithm, mentioned in Chapter II [Evans and Minieka, 1992: 31].

Vertex Cut-set Enumeration. The network analysis model operates on the transformed graph via application of the cut-set enumeration algorithm, mentioned in Chapter II. A general outline of the algorithm is given in the steps below:

1) The vertex adjacency list of the graph of interest, and the two vertices between which separation is desired are entered as input.

2) One of the two separation vertices above is labeled s and is initialized as the only element of a subgraph G_s. The other vertex is labeled t and assigned to set T. The algorithm is started or initially called with the vertices of G_s, on the first call this subgraph contains only the vertex s, and with the elements of the set T, initially set to vertex t.

3) Any vertex adjacent to a vertex in G_s, but not in G_s is considered an element of V_x. If $t \in V_x$ then return out of the current call to this algorithm because no cut-set can contain t except for the cut-set composed only of the element t; otherwise, create a subgraph G_t whose elements are the connected components of t for the graph containing all the vertices and edges of the graph, less those vertices and edges in either V_s or V_x. (Note: The connected components algorithm for this step was derived from the lecture notes of BYU instructors, Sederberg and Venture, 1998, based on a text by Aho and Ullman, 1995. The algorithm is O(mlogn), therefore the overall implementation of this algorithm runs in time O(mnlogn) per vertex cutset.)

4) Let the set Z be composed of those vertices in V_x which have no edge joining them to a vertex in V_t.

5) If Z and T have any elements in common then exit the current call to this algorithm to avoid repetition of cut sets; otherwise continue

6) Add the vertices in Z to those in V_s.

7) The current vertex cut-set, V_c, equals those vertices in V_c, and not in Z.

8) Let Tprime be the empty set.

9) While V_c - T does not equal the empty set do the following loop (descending down the vertex cut-set tree by generating new subgraphs, V_s).

10) Begin Loop:

11) Select and delete any vertex v, an element of V_c - T, add this vertex to the set V_s and add Tprime to T, then recursively call this algorithm with the newly formed V_s and T.

12) Add vertex v to Tprime for blocking purposes.

13) End Loop.

14) Exit. The last step in the program where calls go so that the return out of the current algorithm call occurs.

The vertex cut-sets produced are then converted back to their corresponding network nodes and/or links, and in this converted format, the cut-sets are the candidate target sets. These target sets are then input into the value model for evaluation.

The network analysis model is implemented in a Visual Basic and spreadsheet environment, where Visual Basic modules incorporate the programming required to conduct each part of the process.

The Value Model

The value model takes as input the capabilities and characteristics of the network's nodes and links, and the target sets generated by the network analysis model. The capabilities and characteristics of each network node and link are paired up with their appropriate node or link in each target set. The target sets, can then be scored according

to the model's evaluation measure value functions. Rolling up scores within the value hierarchy model yields an overall value for each target set, and subsequent ranking of the target sets produces a prioritized candidate target set list.

The value model developed is essentially a value hierarchy complete with weights and single dimensional value functions. The evaluation measures and their single dimensional value functions can be developed from relevant literature and the relevant stakeholders, such as C2W planners [Kirkwood, 1997: 21]. Additionally, expert opinion and discussion with individuals familiar with the subject matter can be used in the development process.

The value hierarchy contains the overall fundamental objective for the decision at hand, and the overall fundamental objective characterizes the reason for interest in the decision [Keeney, 1992: 77]. The decision for this thesis is which network components should be targeted for C2-attack in order to achieve network disruption. The fundamental objective is to identify the highest value target set for the C2-attack. Achieving maximum benefits, while keeping costs to the attacker at the minimum, yields the highest value target set. However, these objectives are conflicting and the use of multiobjective decision analysis is the tool to achieve an optimal balance between the objectives. The basis for the value hierarchy was developed through relevant literature review and discussion with individuals familiar with the subject matter [Aegis, 1996; Doyle, 1998]. This proposed target set value hierarchy is shown in Figure 3-3.

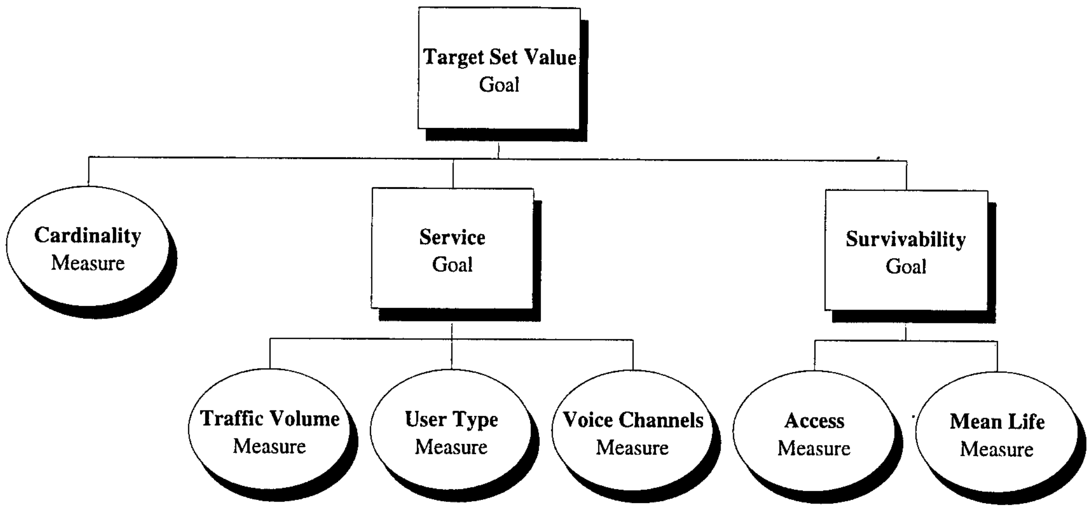

Figure 3-3. Initial Target Set Value Hierarchy

The Multiobjective Value Function. This thesis assumes that there is no uncertainty about the outcome of each alternative. Additionally, it is assumed that the evaluation measures are mutually exclusive, collectively exhaustive and independent. These assumptions allow the measurement of alternatives, target sets in the case of this thesis, to be simpler than when uncertainty is present. Finally, the independence assumption permits the use of an additive value function in the model [Kirkwood, 1997: 239].

A multiobjective value function is an additive value function which is a convenient, yet powerful tool for capturing the value of decision alternatives when tradeoffs between the evaluation measures exist [Kirkwood, 1997: 55, 230]. The form of a multiobjective value function, $v(x)$, is $v(x) = \sum_{i=1}^{n} w_i v_i(x_i)$, where w_i is the global weight

attached to $v_i(xi)$, $\sum_{i=1}^{n} w_i = 1$, and $v_i(x_i)$ is the single dimensional value function of evaluation measure x_i [Kirkwood, 1997: 230].

An analyst at the National Air Intelligence Center suggested the evaluation consideration weights shown in the final hierarchy in Chapter IV. Additionally, the analyst reviewed the proposed single dimensional value functions, shown and described in Appendix A, used in this first cut, proof-of-concept analysis. Given these weights and single dimensional value functions, an overall value of each target set can be determined. The evaluation measures, weights, and value functions are listed and described in Appendix A.

Target Set Evaluations. At this point, a target set's value can be obtained by scoring the set against each evaluation measure in the value hierarchies and combining these results via the multiobjective value function. An example of an evaluation measure might be voice traffic volume across a node or link. This evaluation measure would have a weight associated with it and so would all the other evaluation measures. The score received on each evaluation measure would be multiplied by the evaluation measure's weight and rolled up into the next layer above the evaluation measure. This process would continue until all the evaluation measures were rolled into the overall objective; consequently, the overall target set score would be determined.

The scoring of the alternatives takes place in a spreadsheet environment. The decision maker can enter the target set's scores on the spreadsheet and an automatic calculation of the target sets value will occur, according to the range and shape of the

evaluation measure's value function. This is repeated for each target set. For example, suppose a target set receives a score of fifty on the evaluation measure voice traffic volume. Additionally, assume this evaluation measure's single dimensional value function is linearly increasing on the range of zero to one hundred, with a score of zero receiving a value of zero and a score of one hundred receiving a value of one. A value of .5 would then be assessed for the target set. This process would continue until all target sets had been scored.

Upon determination of all target set values, for all evaluation measures, the target sets are ranked in the Visual Basic environment using a spreadsheet implementation, thus forming a prioritized candidate target set list. Since a network may have many target sets, it is useful for the decision maker if the list of candidate target sets can be truncated at some specified point, in order to form another smaller, but more manageable target set list. This option is available to the decision maker within the Visual Basic environment and provides greater flexibility than if only the entire list of all target sets is given.

Visual Basic and Spreadsheet Tool

A Visual Basic tool has been developed to automate the candidate target set list generation process. User input actions include: placing the network of interest onto a worksheet in Excel; inputting required target characteristics, such as capacity; and entering the weights of the value hierarchy. Changing the default shapes of the single dimensional value functions is an optional user input. All of these actions are accomplished in Excel, through various dialog box prompts and other methods. Given

the prevalence of Excel, its close relationship with Visual Basic, and the graphical user interfacing capabilities of Visual Basic, this tool serves as valuable, user-friendly decision support system. Appendix E contains the Visual Basic module which has instructions for operation of the tool, as well as the screen of the worksheet, Sample, for providing the user with a visual picture of the graph transformation process. Appendices F1 through F9 contain the Visual Basic code of the tool and some of the associated screens.

Summary

The optimal target set for network disruption has been selected using both the scores obtained with the multiobjective value function and application of a cut set generation procedure to the network of interest. These procedures are implemented in a Visual Basic and spreadsheet environment for ease of use by the decision maker and his/her staff. This methodology is a vehicle for the determination of a prioritized candidate target set list in the C2-attack of a telecommunications network.

IV. Results and Analysis

This chapter contains the analysis results for both the network and value models presented in Chapter III.

The Notional Network Model

The notional network (See Figure 4-1) consists of twenty-four nodes and thirty links. Communication disruption is desired between the nodes labeled *Adversary Area 1* and *Adversary Area 2*. Tables 4-1 and 4-2 list the network nodes and links, along with their capabilities and characteristics. The graph representing the notional network has twenty-four vertices and thirty edges prior to transformation into a properly formatted graph for the cut-set enumeration algorithm. After transformation, the graph has fifty four vertices and sixty edges. The graph transformation code is contained in the Visual Basic module shown in Appendix F2.

The cut-set enumeration algorithm applied to the transformed graph yielded 9,079 vertex cut-sets, in under three hours using a personal computer with a Pentium 200 MHz processor. Appendix B contains the first twenty vertex cut-sets, and the entire set is provided on disk. The Visual Basic module containing the code of the algorithm is given in Appendix F4. The potential target sets, composed of network links and/or nodes, corresponding to the first twenty enumerated cut-sets are listed in Appendix C, and the Visual Basic code implementing the cut-set to target set conversion is given in Appendix F5. The remaining target sets are provided on disk.

Table 4-1. Notional Network Nodes

Component	Description[1]	Traffic Volume	User Type	Voice Channels[1]	Access	Mean Life
Node 1	earth station	500	960	8064	2	3
Node 2	central office	500	960	8064	2	3
Node 3	cell site	25	72	832	2	8
Node 4	repeater	500	960	8064	2	1
Node 5	MTSO	25	72	4032	2	10
Node 6	toll central office	350	1072	1800	2	10
Node 7	toll central office	525	1120	11664	2	1
Node 8	repeater	275	720	8064	2	1.5
Node 9	repeater	275	720	8064	2	2
Node 10	repeater	275	720	8064	2	2.5
Node 11	satellite	500	5000	120000	1	5
Node 12	radio relay	150	600	1800	2	7
Node 13	repeater	275	720	8064	2	3
Node 14	repeater	275	720	8064	2	3.5
Node 15	repeater	275	720	8064	2	4
Node 16	repeater	275	720	8064	2	4.5
Node 17	toll central office	275	1120	9864	2	5
Node 18	toll central office	250	600	3600	2	10
Node 19	toll central office	425	1072	1344	2	6
Node 20	repeater	250	480	8064	2	3
Node 21	central office	250	480	8064	2	5
Node 22	earth station	500	1000	1344	2	3
Node 23	Adversary Area 1					
Node 24	Adversary Area 2					

Table 4-2. Notional Network Links

Component	*Description[1]*	*Traffic Volume*	*User Type*	*Voice Channels[1]*	*Access*	*Mean Life*
Link 1-11	microwave	500	5000	120000	1	3
Link 1-23	coaxial/DS-5	500	960	8064	2	18
Link 2-4	coaxial/DS-5	500	960	8064	2	1
Link 2-23	aggregate of twisted pair local loops	500	3200	9600	2	3
Link 3-5	coaxial/DS-4	25	72	4032	2	8
Link 3 -23	microwave	25	100	832	1	8
Link 4-7	coaxial/DS-5	500	960	8064	2	1
Link 5-7	coaxial/DS-4	25	72	4032	2	7.5
Link 6-7	microwave	100	400	1800	1	1
Link 6-19	fiber/DS-3C	350	672	1344	2	6
Link 6-23	fiber/DS-3C	250	1100	1344	2	10
Link 7-8	coaxial/DS-5	275	720	8064	2	1
Link 7-12	microwave	150	600	1800	1	.5
Link 8-9	coaxial/DS-5	275	720	8064	2	1.5
Link 9-10	coaxial/DS-5	275	720	8064	2	2
Link 10-13	coaxial/DS-5	275	720	8064	2	2.5
Link 11-22	microwave	500	5000	120000	1	3
Link 12-18	microwave	150	600	1800	1	5
Link 13-14	coaxial/DS-5	275	720	8064	2	3
Link 14-15	coaxial/DS-5	275	720	8064	2	3.5
Link 15-16	coaxial/DS-5	275	720	8064	2	4
Link 16-17	coaxial/DS-5	275	720	8064	2	4.5
Link 17-18	microwave	100	400	1800	1	5
Link 17-19	microwave	75	400	1800	1	5
Link 17-24	coaxial/DS-5	100	720	8064	2	5
Link 18-20	coaxial/DS-5	250	480	8064	2	3
Link 19-24	fiber/DS-3C	425	1100	1344	2	6
Link 20-21	coaxial/DS-5	250	480	8064	2	5
Link 21-24	aggregate of twisted pair local loops	250	480	9600	2	5
Link 22-24	fiber/DS-3C	500	1000	1344	2	3

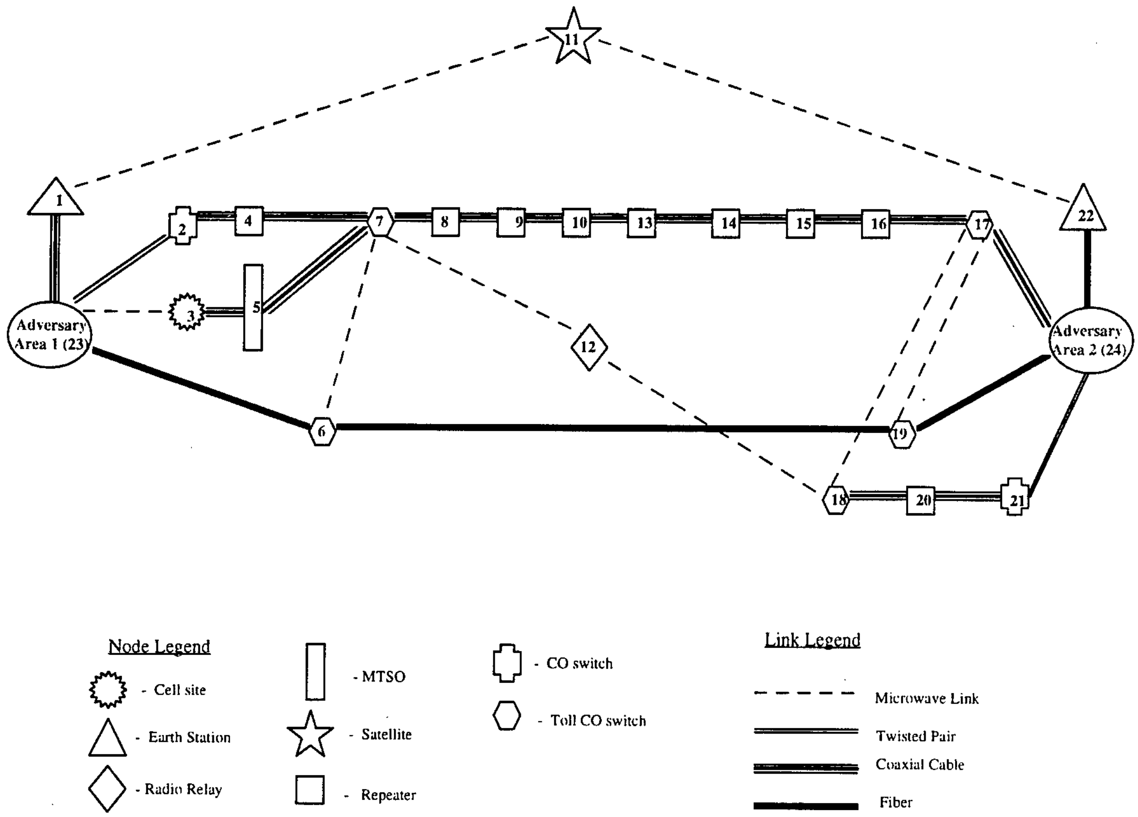

Figure 4-1. Notional Network

Value Model Results

The value model was applied to the target sets generated by the network analysis model. The capabilities and characteristics of each node and link comprising the target sets, given in Tables 4-1 and 4-2, were used to derive the score of each target set. Having determined the target sets' scores, the multiobjective target set value function was used to evaluate each target set. The final value hierarchy, along with its weights, from which the multiobjective value function was derived is shown in Figure 4.2. The hierarchy contains

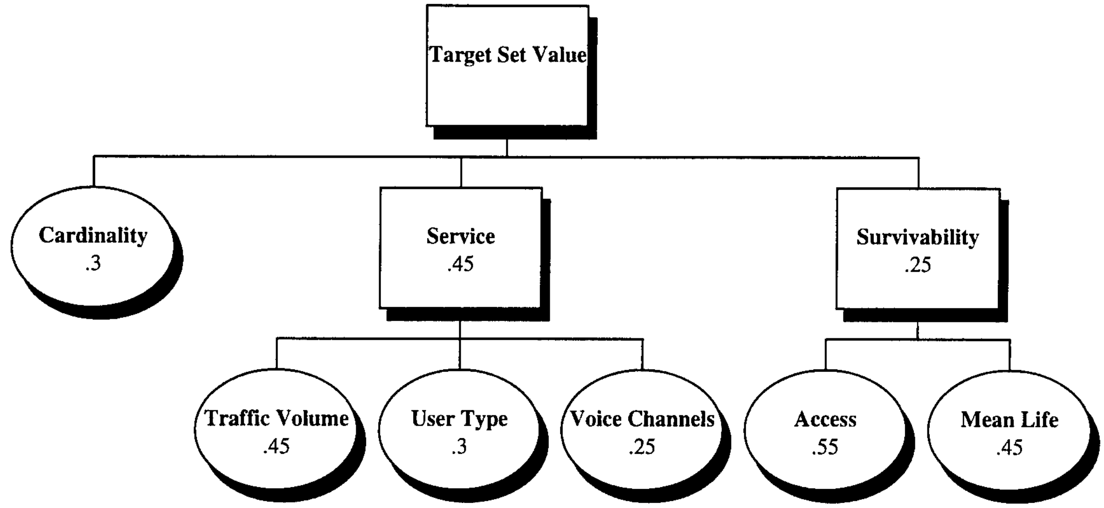

Figure 4-2. Final Target Set Value Hierarchy

six evaluation measures: cardinality, traffic volume, user type, voice channels, access and mean life.

Cardinality measures the value of a target set according to how many targets are in each target set. This evaluation measure essentially serves as an attack cost measure since the number of targets can be translated into the number of missions or number of weapons required for attack of the identified target set. The evaluation measure traffic volume evaluates a target set according to how many voice channels are in use by the adversary; while the voice channels evaluation measure evaluates a target set according to how many voice channels the target set can accommodate (i.e. the voice channels capacity of the target set). User type is an evaluation measure used to calculate the value of a target set by how many of the voice channels within the target set are dedicated for military usage. As such, a user type score will always be greater than or equal to the traffic volume score of a target set and less than or equal to the voice channels score of a

target set. Access evaluates a target set by the number of ways a target set be attacked. Lastly, the evaluation measure mean life determines the value of a target set according to the projected average service life remaining of the targets within the set. A full description of each evaluation measure and its single dimensional value functions is given in Appendix A.

Given the weights and evaluation measures, the multiobjective target set value function is $v(x) = .3v_1(x_1) + .2025v_2(x_2) + .135v_3(x_3) + .1125v_4(x_4) + .1375v_5(x_5) + .1125v_6(x_6)$. The coefficients in the multiobjective value function are global weights. Global weights in the multiobjective value function are calculated by multiplying the weight of the evaluation measure with the weights of those evaluation considerations which are above it in the hierarchy [Clemen, 1996: 557]. For example, the evaluation measure, user type, has a hierarchical weight of 0.3, but a global weight of (0.3)*(0.45) = 0.135, since the service evaluation consideration has a hierarchical weight of 0.45. It is important to note that for evaluation purposes, the highest value any target set can obtain is one, and the lowest value is zero. This value range was determined from the single dimensional value functions and the multiobjective target set value function.

Evaluation measure x_1 is cardinality, x_2 is traffic volume, and so on from left to right in the hierarchy. The same subscript numbering scheme holds for each single dimensional value function, v_i. The range of scores for each value function was calculated using the Visual Basic code shown in Appendix F6, and the capabilities and characteristics of the nodes and links. The ranges of scores, resulting from analyzing the notional network, are shown in Table 4-3.

Table 4-3. Value Function Ranges

EVALUATION MEASURE	RANGE (low, high)
Cardinality (number of targets)	(3, 8)
Traffic Volume (average of voice channels)	(182.14, 483.33)
User Type (average of voice channels)	(612.8, 2486.67)
Voice Channels (average of voice channels)	(2530.67, 44488)
Access (average of ways)	(1.29, 2)
Mean Life (average service life remaining)	(2.63, 10.75)

Each of the 9,079 target sets were scored against each evaluation measure and in accordance with the multiobjective value function, an overall assessment was determined. The VB code used for scoring target set and assessing target set value are given in Appendices F6 and F7, respectively. The scores and values of the 9,079 target sets are provided on disk.

Using the overall target set value, all the target sets were ranked, and the ranked listing of all the target sets comprises the potential target sets for C2-attack of the network. Target sets 7,654 and 5,146 ranked the highest with overall target set values of 0.84, and target sets 8,807 had the lowest rankings with target set values of 0.11. Tables 4-4 and 4-5 show these highest and lowest ranking target sets and their components, respectively.

Table 4-4. Highest Ranking Target Sets

Target Set 7654	*Target Set 5146*
Node 7 - Toll CO	***Node 7 - Toll CO***
Node 11 - Satellite	***Node 11 - Satellite***
Node 6 - Toll CO	***Node 19 - Toll CO***
Value: 0.84	***Value: 0.84***

Table 4-5. Lowest Ranking Target Sets

Target Set 8,802	*Target Set 8,806*	*Target Set 8,807*
Link 6-7: Microwave	***Link 6-7: Microwave***	***Link 6-7: Microwave***
Link 6-23: Fiber DS-3C:	***Link 6-23: Fiber DS-3C:***	***Link 6-23: Fiber DS-3C:***
Link 7-12: Microwave	***Link 7-12: Microwave***	***Link 7-12: Microwave***
Link 17-18: Microwave	***Link 17-18: Microwave***	***Link 17-18: Microwave***
Link 17-19: Microwave	***Link 17-19: Microwave***	***Link 17-19: Microwave***
Link 17-24: Coaxial DS-5	***Link 17-24: Coaxial DS-5***	***Link 17-24: Coaxial DS-5***
Node 1: Earth Station	***Node 22: Earth Station***	***Link 22-24: Fiber DS-3C***
Value: .11	***Value: .11***	***Value: .11***

In order to look at the target sets in more detail, a reduced list of the potential target sets, containing the twenty target sets of highest value, was analyzed, and is shown in Table 4-6. Additionally, the scores of the twenty highest value target sets are given in Table 4-7.

Table 4-6. Reduced List of Potential Targets

Target Set	Components				Value
7654	Node 6 (Toll CO)	Node 7 (Toll CO)	Node 11 (Satellite)		0.84
5146	Node 7 (Toll CO)	Node 11 (Satellite)	Node 19 (Toll CO)		0.84
7653	Node 6 (Toll CO)	Node 7 (Toll CO)	Link 1 to 11 (Microwave)		0.83
7655	Node 6 (Toll CO)	Node 7 (Toll CO)	Link 11 to 22 (Microwave)		0.83
5145	Node 7 (Toll CO)	Node 19 (Toll CO)	Link 1 to 11 (Microwave)		0.83
5147	Node 7 (Toll CO)	Node 19 (Toll CO)	Link 11 to 22 (Microwave)		0.83
8906	Node 7 (Toll CO)	Node 11 (Satellite)	Link 6 to 23 (Fiber)		0.82
6398	Node 7 (Toll CO)	Node 11 (Satellite)	Link 6 to 19 (Fiber)		0.82
8905	Node 7 (Toll CO)	Link 1 to 11 (Microwave)	Link 6 to 23 (Fiber)		0.81
8907	Node 7 (Toll CO)	Link 6 to 23 (Fiber)	Link 11 to 22 (Microwave)		0.81
6397	Node 7 (Toll CO)	Link 1 to 11 (Microwave)	Link 6 to 19 (Fiber)		0.81
6399	Node 7 (Toll CO)	Link 6 to 19 (Fiber)	Link 11 to 22 (Microwave)		0.81
6580	Node 6 (Toll CO)	Node 7 (Toll CO)	Link 1 to 23 (Coaxial)		0.77
4072	Node 7 (Toll CO)	Node 19 (Toll CO)	Link 1 to 23 (Coaxial)		0.77
7832	Node 7 (Toll CO)	Link 1 to 23 (Coaxial)	Link 6 to 23 (Fiber)		0.75
5324	Node 7 (Toll CO)	Link 1 to 23 (Coaxial)	Link 6 to 19 (Fiber)		0.74
2768	Node 7 (Toll CO)	Node 11 (Satellite)	Node 17 (Toll CO)	Link 19 to 24 (Fiber)	0.72
2767	Node 7 (Toll CO)	Node 17 (Toll CO)	Link 1 to 11 (Microwave)	Link 19 to 24 (Fiber)	0.71
2769	Node 7 (Toll CO)	Node 17 (Toll CO)	Link 11 to 22 (Microwave)	Link 19 to 24 (Fiber)	0.71
7722	Node 5 (MTSO)	Node 6 (Toll CO)	Node 11 (Satellite)	Link 2 to 23 (Twisted Pair)	0.71

Table 4-7. Scores for the Twenty Highest Value Target Sets

Target Set	Cardinality Score	Traffic Volume Score	User Type Score	Voice Channels Score	Access Score	Mean Life Score
7654	3	458.33	2477.33	44488	1.67	5.33
5146	3	483.33	2477.33	44488	1.67	4
7653	3	458.33	2477.33	44488	1.67	4.67
7655	3	458.33	2477.33	44488	1.67	4.67
5145	3	483.33	2477.33	44488	1.67	3.33
5147	3	483.33	2477.33	44488	1.67	3.33
8906	3	425	2486.67	44336	1.67	5.33
6398	3	458.33	2344	44336	1.67	4
8905	3	425	2486.67	44336	1.67	4.67
8907	3	425	2486.67	44336	1.67	4.67
6397	3	458.33	2344	44336	1.67	3.33
6399	3	458.33	2344	44336	1.67	3.33
6580	3	458.33	1130.67	7176	2	9.67
4072	3	483.33	1130.67	7176	2	8.33
7832	3	425	1140	7024	2	9.67
5324	3	458.33	997.33	7024	2	8.33
2768	4	431.25	2145	35718	1.75	4.25
2767	4	431.25	2145	35718	1.75	3.75
2769	4	431.25	2145	35718	1.75	3.75
7722	4	343.75	2336	33858	1.75	7

Node and Link Composition. Figure 4-3 shows the values of the targets sets in the reduced list of potential targets, according to how much the first tier below the overall evaluation consideration contributes to the target set value. This first tier is composed of the evaluation measure cardinality, and the evaluation considerations, service and survivability. Figure 4-4 graphically represents the ranking of the target sets and their composition by the six value model evaluation measures. Appendix D provides a table showing the numerical values of each measure for the top twenty target sets.

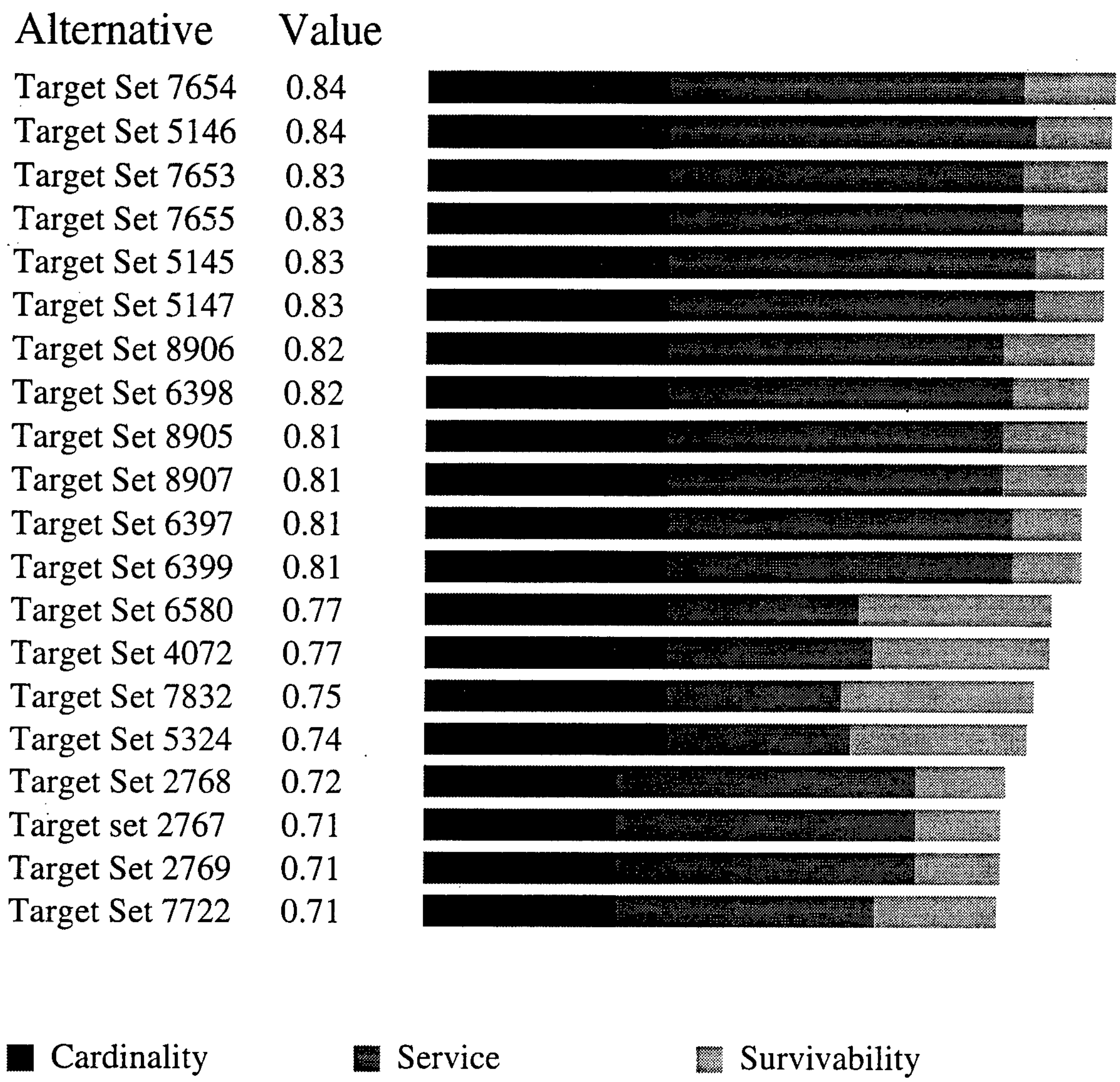

Figure 4-3. First Tier Target Set Value Ranking

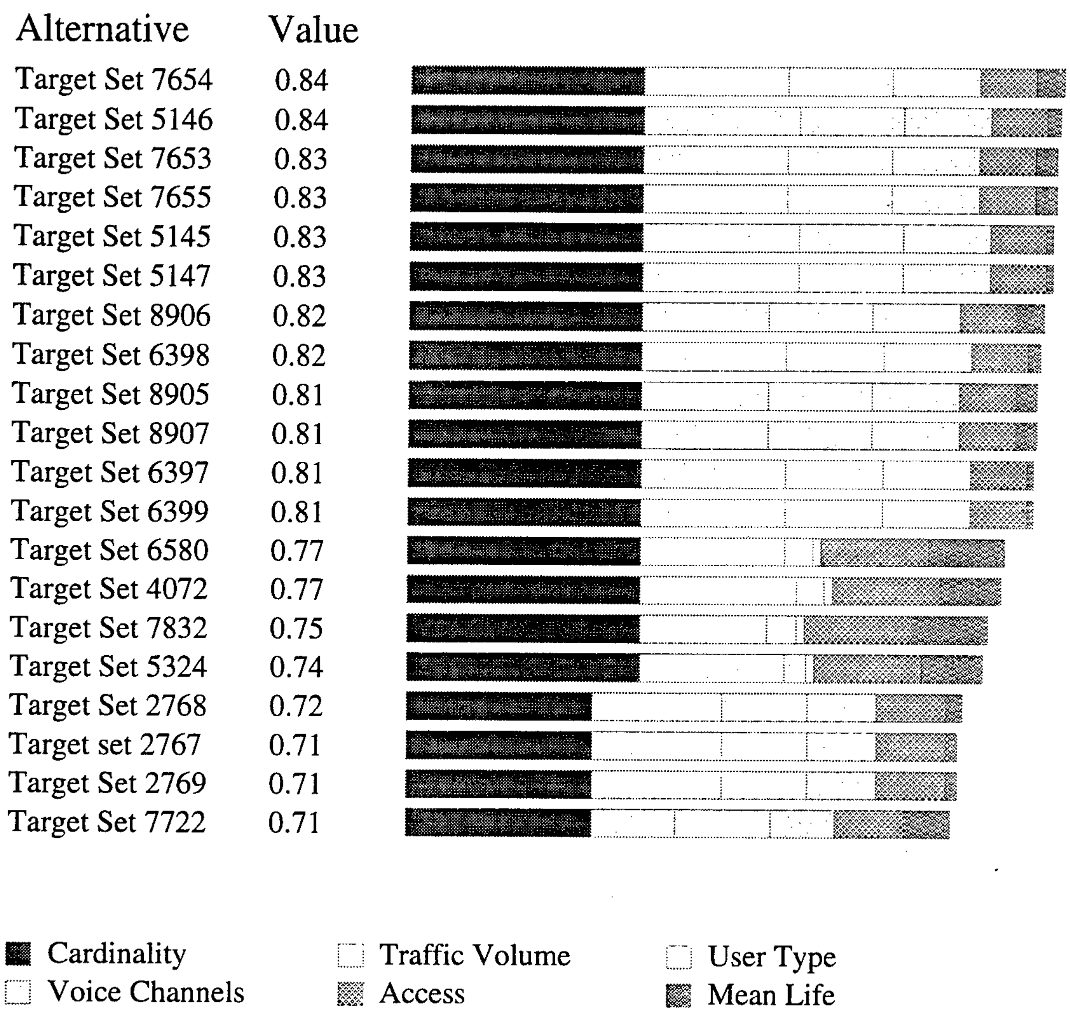

Figure 4-4. Target Set Value Ranking by Evaluation Measures

From Figure 4-4, the contribution of cardinality in the top twenty target sets stands out as the most important impact on the target sets, followed by traffic volume. This fact is expected in light of the weights for these two evaluation measures; where cardinality has the highest weight and traffic volume the second highest weight. From Figure 4.4

and Appendix D, seven target sets emerge as non-dominated target sets. These include the following sets: 7654, 5146, 8906, 6580, 4072, 2768, and 7722.

Sensitivity Analysis. In an effort to gain insight into the importance the weights have on the reduced list of potential target sets, sensitivity analysis was conducted by swinging the weights for each evaluation measure. It is important to note that as an evaluation measure's weight approaches zero, that evaluation measure essentially becomes a non-player in the decision at hand, while a weight that approaches one makes the problem less of a decision analysis problem and more of an optimization effort for the high weight evaluation measure. In this phase of the analysis, only the top five of the final potential target set list were analyzed for sensitivity to changes in the evaluation measure weights.

Figure 4-5 displays the sensitivity of the five target set rankings to the weight of the evaluation measure cardinality. Figure 4-5 and Table 4-8, show that target sets 7654 and 5146 remain the top two target sets as the weight on the evaluation measure cardinality is increased from zero toward one, but eventually all five target sets converge to the same value, because the five target sets received the same score for cardinality.

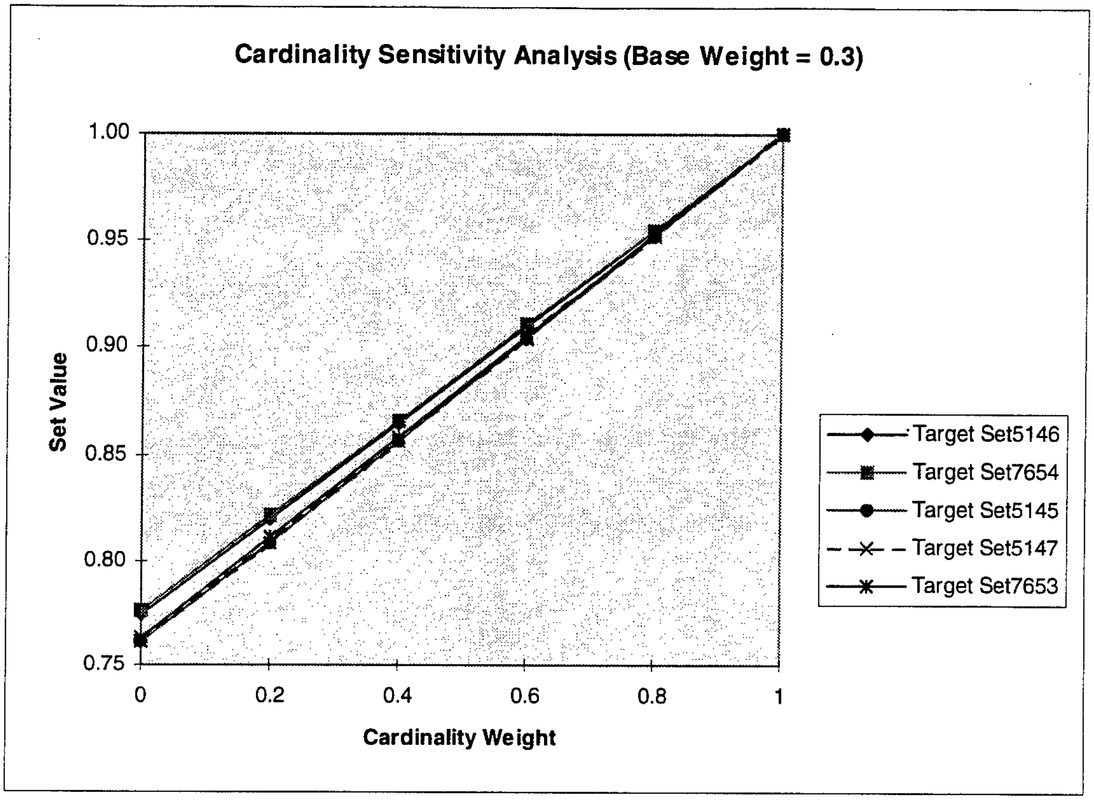

Figure 4-5. Cardinality Weight Sensitivity Graph

Table 4-8. Sensitivity Analysis Table for Cardinality Evaluation Measure

Weight	.3	0	0.2	0.4	0.6	0.8	1
Target Set 5146	0.84	0.77	0.82	0.86	0.91	0.95	1.00
Target Set 7654	0.84	0.78	0.82	0.87	0.91	0.96	1.00
Target Set 5145	0.83	0.76	0.81	0.86	0.90	0.95	1.00
Target Set 5147	0.83	0.76	0.81	0.86	0.90	0.95	1.00
Target Set 7653	0.83	0.76	0.81	0.86	0.91	0.95	1.00

Target set value was very sensitive to the weight of the traffic volume evaluation measure as shown in Figure 4-6. Table 4-9 verifies that the rankings were sensitive to even small changes in the weight of the traffic volume evaluation measure. A small

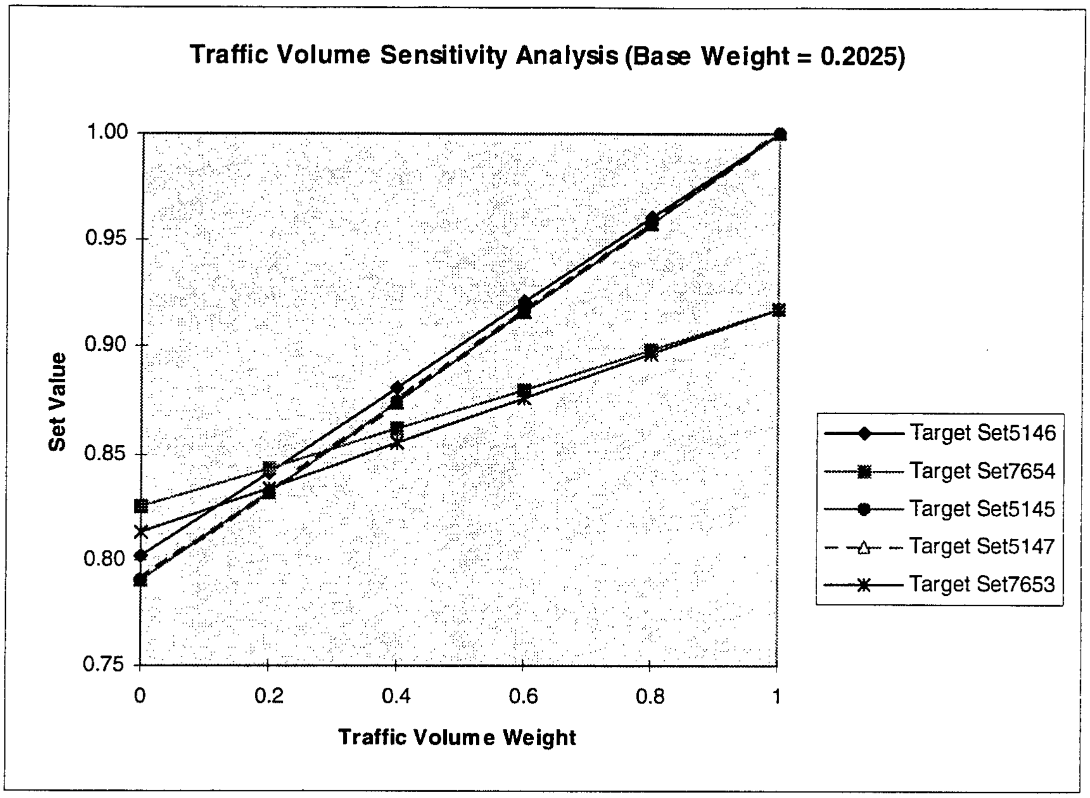

Figure 4-6. Traffic Volume Sensitivity Graph

Table 4-9. Sensitivity Analysis Table for Traffic Volume Evaluation Measure

Weight	.2025	0	0.2	0.4	0.6	0.8	1
Target Set 5146	0.84	0.80	0.84	0.88	0.92	0.96	1.00
Target Set 7654	0.84	0.82	0.84	0.86	0.88	0.90	0.92
Target Set 5145	0.83	0.79	0.83	0.87	0.92	0.96	1.00
Target Set 5147	0.83	0.79	0.83	0.87	0.92	0.96	1.00
Target Set 7653	0.83	0.81	0.83	0.85	0.88	0.90	0.92

increase in the weight of the traffic volume evaluation measure results in target set 7,654 moving down from first place to third place, and a small decrease in the weight of the evaluation measure causes target set 7,653 to surpass target set 5,146, originally tied for the highest value of all five target sets. Further research into the certainty of this

evaluation measure's weight is certainly required in order to make the proper target set selection. Additionally, swing-weighting among the five target sets may be necessary to help determine the appropriate weight for the measure traffic volume.

The sensitivity trend seen in the evaluation measure cardinality is once again exhibited in the user type evaluation measure. Figure 4-7 and Table 4-10 show that the user type weight has minimal impact on the rankings, even at the extreme ranges of the weights, and there exists convergence of the rankings as the weight approaches one. This convergence is expected since the five target sets had the same user type score.

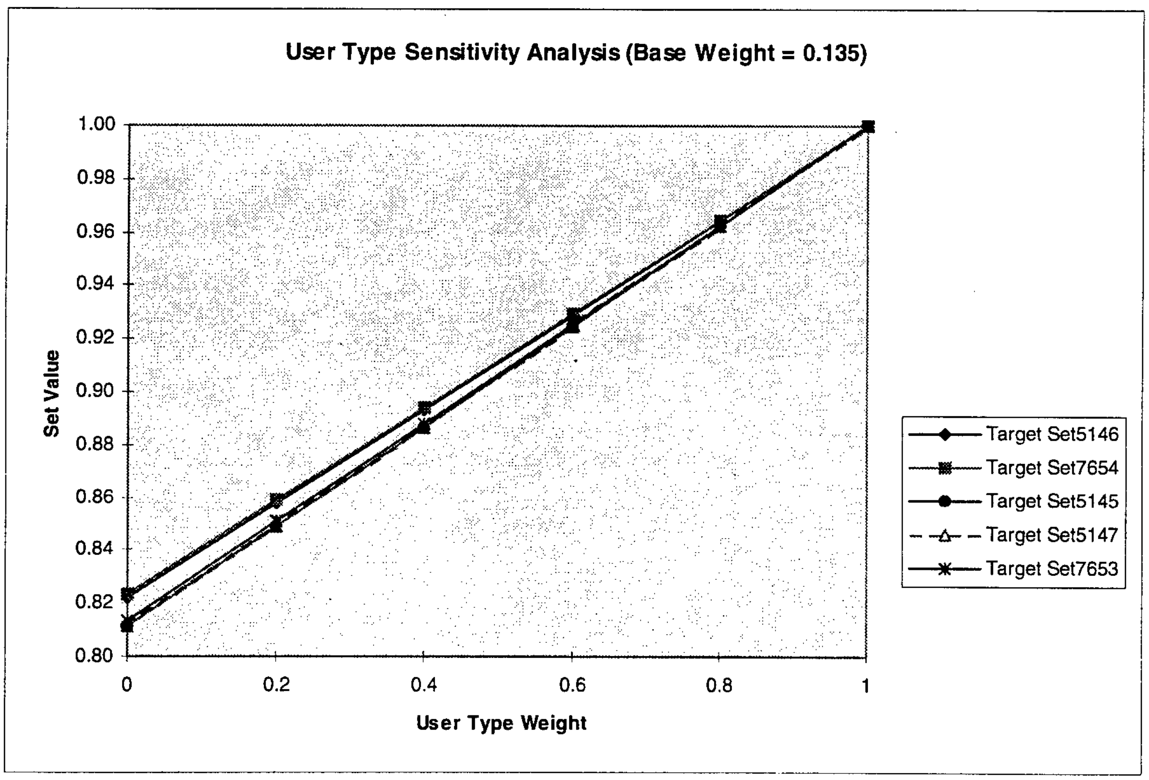

Figure 4-7. User Type Sensitivity Graph

Table 4-10. Sensitivity Analysis Table for User Type Evaluation Measure

Weight	.135	0	0.2	0.4	0.6	0.8	1
Target Set 5146	0.84	0.82	0.85	0.89	0.92	0.96	1.00
Target Set 7654	0.84	0.82	0.85	0.89	0.92	0.96	1.00
Target Set 5145	0.83	0.81	0.84	0.88	0.92	0.96	1.00
Target Set 5147	0.83	0.81	0.84	0.88	0.92	0.96	1.00
Target Set 7653	0.83	0.81	0.85	0.88	0.92	0.96	1.00

Figure 4-8 and Table 4-11 show that the target set rankings are not very sensitive to the weight of the evaluation measure, voice channels. Additionally, the convergence trend seen in the user type evaluation measure is seen again for the voice channels evaluation measure. Once again, this trend is caused by the five target sets having the same score.

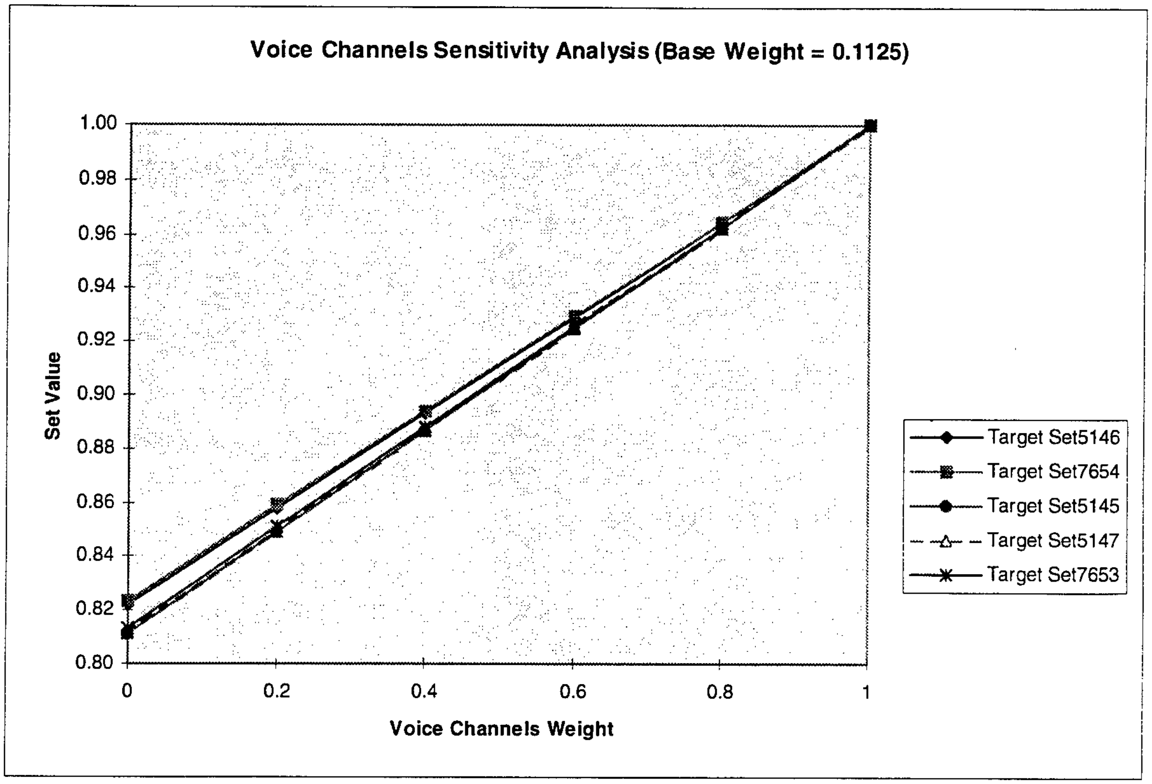

Figure 4-8. Voice Channels Sensitivity Graph

Table 4-11. Sensitivity Analysis Table for Voice Channels Evaluation Measure

Weight	.1125	0	0.2	0.4	0.6	0.8	1
Target Set 5146	0.84	0.82	0.86	0.89	0.93	0.96	1.00
Target Set 7654	0.84	0.82	0.86	0.89	0.93	0.96	1.00
Target Set 5145	0.83	0.81	0.85	0.89	0.92	0.96	1.00
Target Set 5147	0.83	0.81	0.85	0.89	0.92	0.96	1.00
Target Set 7653	0.83	0.81	0.85	0.89	0.93	0.96	1.00

Figure 4-9 shows the sensitivity of the top five target sets to changes in the weight of the evaluation measure, access. Figure 4-9, along with Table 4-12, indicates that there are no changes in the target set rankings, even when the weight of the access

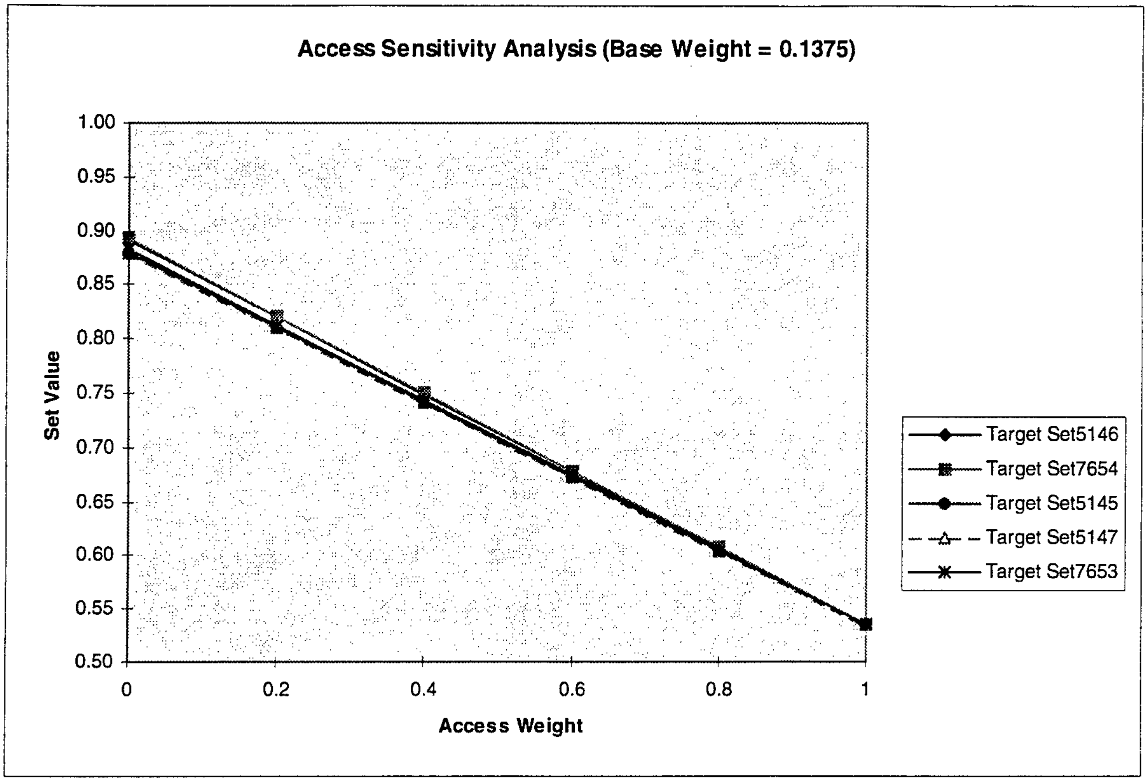

Figure 4-9. Access Weight Sensitivity Graph

Table 4-12. Sensitivity Analysis Table for Access Evaluation Measure

Weight	.1375	0	0.2	0.4	0.6	0.8	1
Target Set 5146	0.84	0.89	0.82	0.75	0.68	0.61	0.54
Target Set 7654	0.84	0.89	0.82	0.75	0.68	0.61	0.54
Target Set 5145	0.83	0.88	0.81	0.74	0.67	0.60	0.54
Target Set 5147	0.83	0.88	0.81	0.74	0.67	0.60	0.54
Target Set 7653	0.83	0.88	0.81	0.74	0.67	0.60	0.54

evaluation measure is swung from zero to one hundred percent. The convergence trend seen in previous evaluation measures is exhibited again since the five target sets all received the same score for access.

Figure 4-10 and Table 4-13 show the sensitivity of the five target sets to changes in the weight of the mean life evaluation measure. For this evaluation measure, there is indeed an impact on the ranking of the top five target sets when the mean life weight is shifted. In fact, the ranking is extremely sensitive to the mean life evaluation measure, as a small variation either way from the current mean life weight causes the rankings to change. An almost complete reversal of the ranks occurs when the mean life weight is shifted from it current amount. Initially, target sets 7,654 and 5,146 are the top two target sets, but swinging the weight from .1125 to zero causes target sets 5,145 and 5,147 to surpass target set 7,654 in value. Increasing the weight from its base of .1125 causes target set 5,146 to drop in value and target set 7,653 to increase in value, such that target set 7,653 is second only to target set 7,654. This sensitivity indicates that more in-depth analysis should be conducted for the weight of this evaluation measure; specifically, the certainty of the weight should be examined. As was the case with the measure, traffic volume, swing weighting may be needed for proper mean life weight determination.

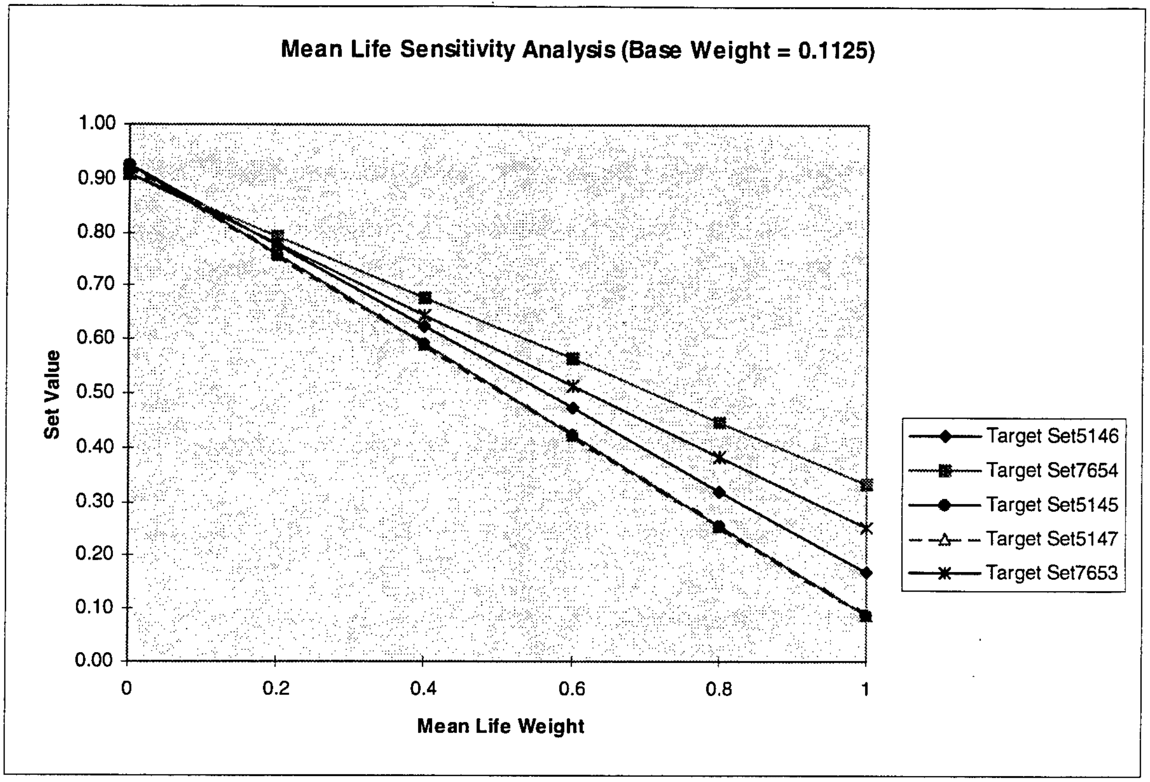

Figure 4-10. Mean Life Sensitivity Graph

Table 4-13. Sensitivity Analysis Table for Mean Life Evaluation Measure

Mean Life	.1125	0	0.2	0.4	0.6	0.8	1
Target Set 5146	0.84	0.93	0.78	0.62	0.47	0.32	0.17
Target Set 7654	0.84	0.91	0.79	0.68	0.56	0.45	0.33
Target Set 5145	0.83	0.93	0.76	0.59	0.42	0.25	0.09
Target Set 5147	0.83	0.93	0.76	0.59	0.42	0.25	0.09
Target Set 7653	0.83	0.91	0.78	0.65	0.51	0.38	0.25

The sensitivity graphs give insight into how sensitive the target set values and their rankings are to changes in the various evaluation measure weights for the particular notional network scenario used in this analysis. Due to the fact that the top five target sets had the same scores for the evaluation measures, access, cardinality, user type, and

voice channels, the weights attached to those evaluation measures had negligible impact on the rankings of the target sets. However, since the target set scores for the evaluation measures, mean life and traffic volume, differed, the five target sets were sensitive to those evaluation measure weights. In light of how sensitive the top five target sets are to the weights of the mean life and traffic volume evaluation measures, further investigation into the certainty of those weights is required to ensure good target set decisions are made. Swing weighting may be the tool to assist in determining the proper weight for these two measures. Additionally, since the weights of these two measures have such an impact on the top five target set rankings, elimination of all the other measures and then a re-weighting of the mean life and traffic volume measures may be needed. Such analysis could aid a decision maker in differentiating between target sets based solely on these two measures, which initially had a high impact on the rankings.

Additional Analysis. An additional result of the analysis was noted, namely, target recurrence, or persistence, seen from the data in Table 4-14. Figure 4-11 shows a chart of node persistence for all the nodes in the final list of potential target sets, while Figure 4-12 shows link persistence within the final list. Node 7, a toll central office, was the most persistent target in the final list of potential targets, contained in nineteen of the twenty target sets. Figure 4-1 suggests that one reason for this persistency is the large number of paths going through node 7, resulting in less additional targets required to form a cut-set. This explains why node 7, while a key target in the high ranking cut-sets, was the least recurring target out of all 9079 target sets. Targeting node 7 has a high impact because

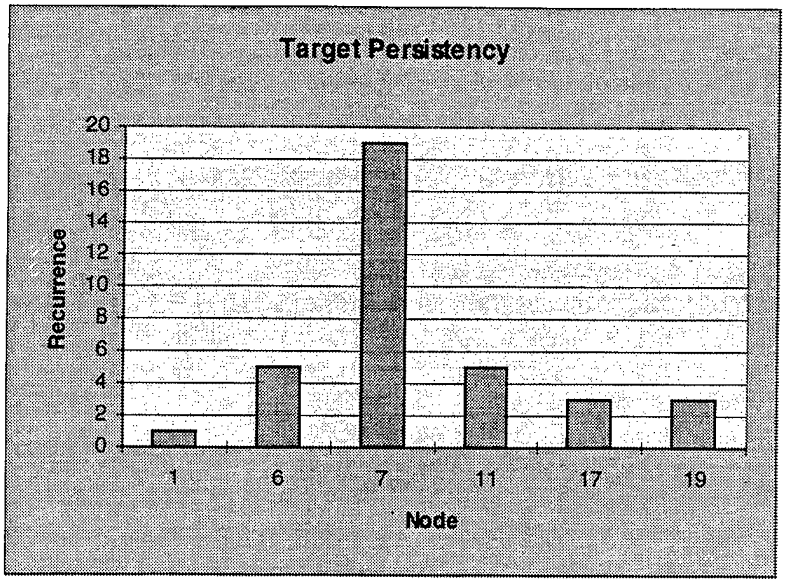

Figure 4-11. Node Target Persistency

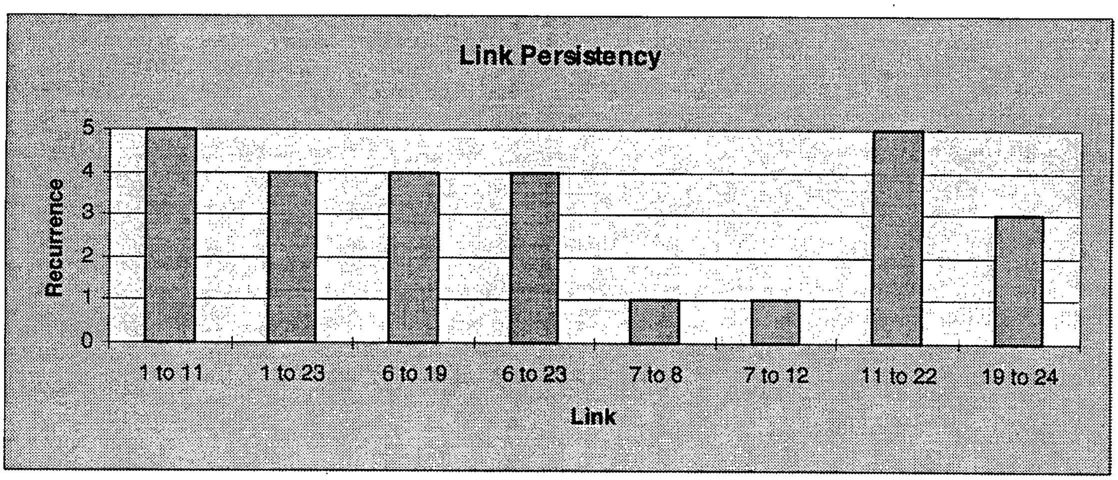

Figure 4-12. Link Target Persistency

the target set cardinality was weighted the highest among all evaluation measures. Additionally, node 7 had the largest capacity of any network node and also had a relatively large voice channel capacity. In Figure 4-11, it is noteworthy that two of the

most persistent nodes, node 7 and 11, have very high relative traffic volumes. Node 7 is a toll central office with a traffic volume of 525 voice channels, while node 11 is the satellite with a traffic volume of 500 voice channels. Node 6, which is a fairly persistent node, is distinctively different from the other persistent nodes 7 and 11, namely the mean life of node 6 is second only to the mean life of link 1 to 23. The mean lives of nodes 7 and 11 are one and five years, respectively; while node 6 has a mean life of ten years. Therefore, even though the traffic volume of node 6 is lower than that of nodes 7 and 11, its long mean life seems to counteract this detraction and contribute, at least in part, to its persistency within the final list of potential target sets.

From Figure 4-12, it is seen that the most persistent links within the final target sets are the links between the earth stations and the satellite, namely, links 1 to 11 and 11 to 22. Closer analysis reveals that for the evaluation measures traffic volume, user type and voice channels, these links score as high or higher than any other links in the network, resulting in a value as great as or greater than any other links in the network.

Visual Basic and Spreadsheet Tool

The Visual Basic tool developed is essentially self-driven. An instruction worksheet, titled Intro-help, is provided in the tool and is also given in Appendix E. A seven button tool bar is the user interface throughout the target nomination process. Each button is described in the instruction worksheet.

Summary

The target set nomination process can be enhanced by the methodology presented in this chapter. Given the notional network of Figure 4-1, a final list of candidate target sets was determined, and this list contained the highest ranking target sets, for the assumptions of this effort; since no target sets were excluded and the value model adequately captured the value of each target set.

The sensitivity analysis conducted on the five target sets, with the highest value, identified the mean life and traffic volume evaluation measure weights as critical in assessing the rankings of the top five target sets. Future study in this area should include uncertainty analysis of these two weights.

The Visual Basic tool provided a convenient method for carrying out much of the target set nomination list process. Given this tool, the methodology described in this chapter, and the analysis conducted, the proof of concept is complete, and further research into various aspects of this problem can continue from here.

V. Conclusions and Recommendations

This chapter outlines the overall results of this effort and provides recommendations for future, related research.

Overview

This effort accomplished the goal of providing a methodology whereby C2-attack target sets can be generated, and then quantitatively measured for their value in achieving an overall objective of maximizing communication disruption between two adversaries. Additionally, a Visual Basic tool, to assist in pursuance of the methodology, was developed. Graphical and textual displays were produced in order to assist in determining the appropriate target nominations.

Research Results

The analysis conducted on the notional network proved that the concepts of generating network target sets to be attacked and subsequent measurement of these target sets to nominate the most valuable target sets are feasible and warrant continued research.

Given a communication network of interest, the network analysis model provides a convenient way to generate target sets through cut set enumeration. The target set value model continues with the generated target sets and implements value focused thinking methods based upon decision maker preferences. As such, a logical and methodical assessment of the value of network target sets can be accomplished. The output of these

models can be used to nominate target sets and also used as input for a weapons to target assignment model or some other analytical tool.

Limitations of the Study

The moderate performance of the target set generation program currently limits the real world, "hot spot" application of this tool to small communication networks; however, with program refinements and computer processors speeds continually increasing, the tool could be useful, not only in longer term planning, but also in quick turn situations.

This tool assumes deterministic network features and non-dynamic network activity; however, the methodology is still certainly valid.

Recommendations for Future Research

As mentioned above, the target set generation program is not extremely quick in its processing time. Refinement of the algorithm or replacement with a better order algorithm would certainly increase the real world applicability of this model. A heuristic approach has been put forth for application when time is of the essence. The idea is to have the program judiciously choose the available targets to enter a target set. Optimality is not guaranteed, but continual improvement is, and the length of the program run would probably be much more appealing.

The value model composed was a first cut value model. As such, additional input from knowledgeable agencies would greatly increase the probability of developing a

model that completely captures the essential elements of the target set selection decision, while maintaining independence among the elements.

The value functions for the measure were assumed to be linear and served well for the purposes of proving the effort's concepts. Non-linear value functions are probably warranted for some of the measures, if not all, in this effort. Elicitation of the exact shape and ranges of the value functions from relevant stakeholders would certainly increase model viability.

Notional network scenarios can be difficult to generate, especially for the purpose of illustrating concepts and methodologies, and as such, real world networks are often preferred. While the fabricated network of this effort sufficed to prove various concepts, more diverse network scenarios could enhance the insight gained from this effort's models. One possible approach to the creation of such scenarios is the random generation of various network components and topologies, which meet various design criteria. Application of this effort's models to several such generated networks, is likely to yield further insight.

Attachment of a weapons to target assignment model to the tool would aid in the effort to provide a more inclusive tool. Additionally, compatibility issues among databases and additional or increased user friendly macros would allow for implementation of the tool by several agencies.

Conclusions

The methodology provided and carried out in this effort allows for the selection of potential target sets, which have the highest value for the network disruption objective at hand. Additionally, the Visual Basic tool applied to a notional network demonstrated the actual application of the models; therefore proving that the models were operational and plausible. Despite the limitations of this effort, the underlying concepts are proven and with further refinement can be implemented in an operational environment to yield sound results in the target planning process.

Bibliography

Aegis Research Corporation. *IW-C2W Weapon/Target Prioritization/Selection.* 3 June 1996

Aho, Alfred V. and Jeffrey D. Ullman. *Foundations of Computer Science.* New York, New York: W. H. Freeman and Company, 1995.

Alliance for Telecommunications Industry Solutions. T1A1.2 Working Group on Network Survivability Performance. *A technical report on network survivability performance.* Washington, D.C.: The Alliance, 1993.

Bazaraa, Mokhtar S., John J. Jarvis, and Hanif D. Sherali. *Linear Programming and Network Flows.* New York, New York: John Wiley and Sons, Inc., 1977.

Biegel, John E. "Determination of Tie Sets and Cut Sets for a System Without Feedback," *IEEE Transactions on Reliability, Vol. R-26.; No. 1*: 39-42 (April 1997).

Bunger, Ronald E. 38th Engineering Installation Group, Tinker AFB, OK. Telephone interview. 9 March 1998.

Clemen, Robert T. *Making Hard Decisions: An Introduction to Decision Analysis,* 2d ed. Belmont, California: Duxbury Press, 1996.

Couch, Leon W. Couch II *Modern Communication Systems: Principles and Applications.* Englewood Cliffs, New Jersey: Prentice Hall, 1995

Davis, Christine C. *A Methodology for Evaluating and Enhancing C4I Networks.* MS Thesis, AFIT/ENS/GOR-97M

Department of Defense, Joint Chiefs of Staff. (21 January 1997). *First Draft, Joint Publication 3-13: Joint Doctrine for Information Operations.*

Department of Defense, Joint Chiefs of Staff. (7 February 1996). *Joint Publication 3-13.1: Joint Doctrine for Command and Control Warfare (C2W).* Available Internet: www.defenselink.mil

Department of Defense, Joint Chiefs of Staff. (30 May 1995). *Joint Publication 6-0: Doctrine for Command, Control, Communications, and Computer (C4) Systems Support to Joint Operations.* Available Internet: www.defenselink.mil

Department of the Air Force. (1993). *AFI 14-207: Air Force Targeting*. Available Internet: http://afpubs.hq.af.mil/elec-products/

Doyle, Michael P. *A Value Focused Thinking Approach to Offensive Information Operations (Secret).* AFIT/ENS/GOR98M.

Evans, James R. and Edward Minieka. *Optimization Algorithms for Networks and Graphs*, 2d ed. New York: Marcel Dekker, Inc., 1992.

Flood, J. E. *Telecommunications Networks.* Stevenage, Herts, England: Peter Peregrinus Ltd., 1975.

Ford, L. R. Jr. and D. R. Fulkerson. *Flows in Networks.* Princeton, New Jersey: Princeton University Press, 1962.

Frank, Howard and Ivan T. Frisch. *Communication, Transmission and Transportation Networks.* Reading, Massachusetts: Addison-Wesley Publishing Company, 1971.

http://www.crtc.gc.ca/eng/telecom/decision/1996/d9613_0.txt, 1996.

"Infrastructural Warfare Situation Report," *Journal of Infrastructural Warfare*: (1997). Available Internet: www.iwar.org

Intelsat, http://www.intelsat.com/cmc/info/is6.htm, 6 January 1998.

Keeney, Ralph L. *Value-Focused Thinking: A Path to Creative Decision Making.* Cambridge, Massachusetts: Harvard University Press, 1992.

Kirkwood, Craig W. *Strategic Decision Making: Multiobjective Decision Analysis with Spreadsheets*. Belmont: Duxbury Press, 1997.

Martin, James. *Telecommunications and the Computer*. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1976

Miller, Michael J. and Syed V. Ahamed. *Digital Transmission Systems and Networks.* Rockville, Maryland: Computer Science Press, Inc., 1988.

Microsoft Corporation. *Visual Basic User's Guide, Microsoft Excel Version 5.0: Automating, Customizing, and Programming in Microsoft Excel with the Microsoft Visual Basic Programming System, Applications Edition.* 1993-1994.

Patvardhan C., V.C. Prasad, and V. Prem Pyara. "Vertex Cutsets of Undirected Graphs," *IEEE Transactions on Reliability, Vol. 44., No. 2*: 347-353 (June 1995).

Prasad V.C., V. Sankar, and K.S. Prasaka Rao. "Generation of Vertex and Edge Cutsets," *Microelectronics And Reliability, Vol. 32, No. 9*: 1291-1310 (1992).

Saadawi, Tarek N., Mostafa H. Ammar, and Ahmed El Hakeem. *Fundamentals of Telecommunication Networks*. New York: John Wiley and Sons, Inc., 1994.

Sederberg, Thomas W. and Dan Ventura. Lecture Notes, CS236, Foundations of Computer Science II. College of Physical and Mathematical Sciences, Brigham Young University, Provo UT. Winter 1998. Available Internet: http://www.cs.byu.edu/courses/cs236/lectures/GRAPHS3/Graphs3.html.

Appendix A. Target Set Value Hierarchy, Measures and Value Functions

This appendix shows the final cut target set value hierarchy along with its weights (See Figure A-1). Additionally, each measure in the hierarchy is explained, and the single dimensional value function for each measure is also presented.

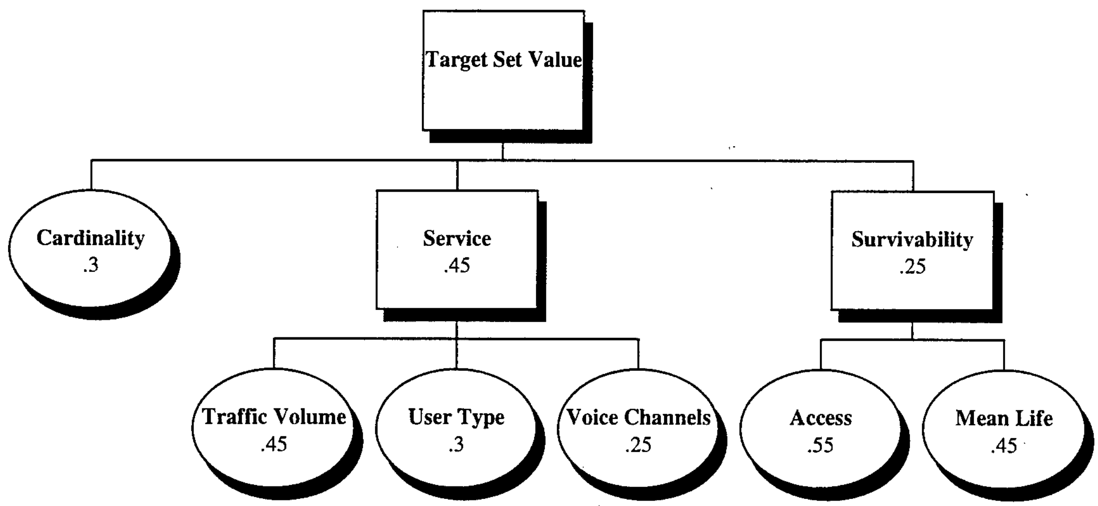

Figure A-1. Target Set Value Hierarchy with Weights

Cardinality

The first evaluation measure is target set cardinality. This measures the number of targets in the target set. Given the assumption that each target will require at least one weapon, the fewer the number of targets an attacker must incapacitate, the better, regardless of the attack weapon used. Figure A-2 shows the single dimensional value function used to describe this measure. This function shows the value of a target set relative to the number of targets composing the set. The horizontal axis' scale is

determined as follows: a network analysis is performed on the network of interest yielding candidate target sets; out of all the target sets, the cardinality of the set with the fewest targets is labeled a_c and receives a value of one; the cardinality of the set with the most targets is labeled b_c and receives a value of zero. These cardinalities are subject to the network of interest. Note that the cardinality of any target set in a non-trivial network must be greater than zero.

The mathematical representation of a target set's score is: score = |{target set}|, where |{set}| is the cardinality of the set; a_c = min{target set *cardinality* scores} and b_c = max{target set *cardinality* scores}.

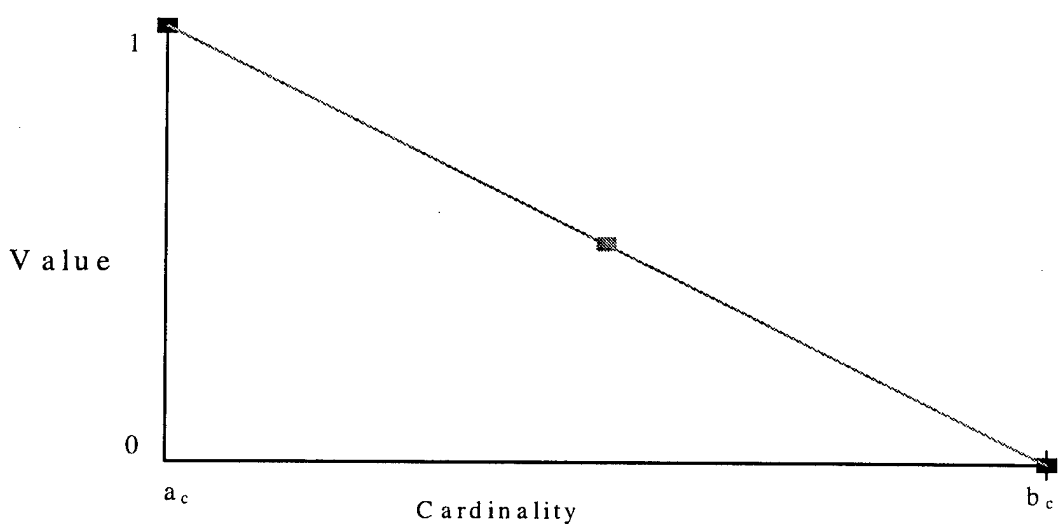

Figure A-2. *Cardinality* Single Dimensional Value Function

Service

This evaluation consideration consists of three measures: *traffic volume*, *user type* and *voice channels*. The objective of this consideration is to choose the target set

which corresponds to the components with the highest traffic volume and capacity, while affecting primarily military personnel.

Traffic volume: This measure identifies the value of a target set according to the adversary's traffic flow amount through the target set. The more traffic that flows through the individual targets comprising the set, the more valuable the targets are to the user and thus to the attacker. If the target is a link, then the traffic volume is equal to the number of voice channels being utilized within the link. If the target is a node, the traffic volume is equal to the number of voice channels being utilized across the links incident to that node, such that conservation of flow is maintained at that node.

The *traffic volume* measure is different from that of capacity, which is taken into account by the measure, *voice channels*. The single dimensional value function is shown in Figure A-3. The horizontal scale's unit of measurement is number of voice channels. The range of the scale is determined by analysis of all the target sets. Each target set is examined individually and the targets within the set have a certain amount of traffic flowing through them (i.e. a number of voice channels in use). The mean of all the targets' voice channels in use is equated to the set's traffic volume score. The highest score of all the target sets is labeled b_{tv}, receiving a value of one, and the lowest score of all sets is labeled a_{tv}, receiving a value of zero. A target set's score is mathematically equivalent to:

$$\frac{\sum_{\forall i \in \text{target set}} \text{traffic volume of target } i}{\left|\{\text{target set}\}\right|}.$$

Additionally,

a_{tv} = min{target set *traffic volume* scores}

b_{tv} = max{target set *traffic volume* scores}

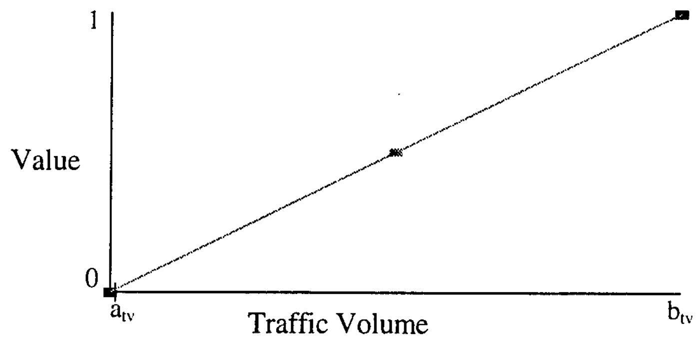

Figure A-3. *Traffic Volume* Single Dimensional Value Function

User Type: This measure determines the value of a target set relative to the primary user of the target. It is feasible for the adversary to use civilian communication components for their purposes and thus a necessity may exist to attack such components; however, it is preferred to disrupt or degrade *military* communications by attacking components that primarily service the adversary. A target set is scored by examining each target in the set and determining the number of voice channels dedicated to the military, for each target. If the target is a link, then the user type amount is equivalent to the number of the link's voice channels allocated to the military. When the target is a node, the user type amount is equal to the number of voice channels, dedicated to the military, across the links incident to the node, such that conservation of flow is maintained at that node. Additionally, the node's incident links' user type amounts are not permitted to be greater than their military voice channel allotments.

The mean of the number of voice channels dedicated to the military, for all the targets in the set, is the set's score. The range of the horizontal axis is from a_{ut} to b_{ut}, where a_{ut} corresponds to the highest mean of all the target sets, and b_{ut} corresponds to the lowest mean of all the sets. The scale's unit is number of voice channels. The value of a_{ut} is zero and the value of b_{ut} is one. The associated, linear single dimensional value function is shown in Figure A-4. A mathematical representation of a target set's score is :

$$\frac{\sum_{\forall i \in \text{target set}} \text{target } i\text{'s user type amount}}{\left|\{\text{target set}\}\right|}$$

Additionally,

$$a_{ut} = \min\{\text{target set } \textit{user type} \text{ scores}\}$$

$$b_{ut} = \max\{\text{target set } \textit{user type} \text{ scores}\}$$

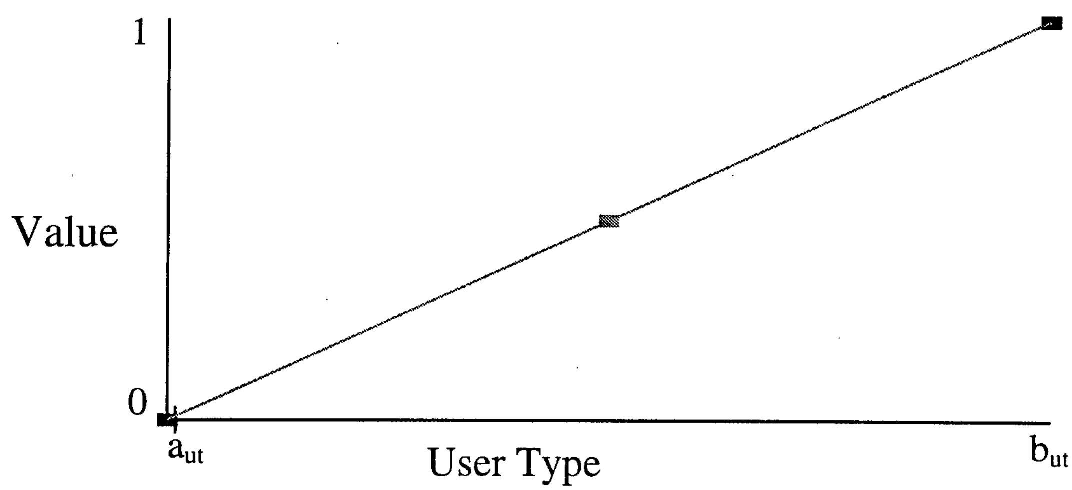

Figure A-4. *User Type* Single Dimensional Value Function

Voice channels is an evaluation measure that measures the value of a target set according to the number of voice-frequency (VF) channels a target set can accommodate. For a link, the voice channels amount is equal to the VF channel capacity for a link. For a node, the voice channels amount is equal to the VF channel capacity of the links incident to the node, such that conservation of flow is maintained at that node. Additionally, the node's incident links' voice channels amounts are not permitted to be greater than their VF channel capacities.

The mean of the number of VF channels for all the targets in a set is the target set score. The range of the scale is determined by examining the network of interest. The largest mean of all the target sets is the maximum score, b_{vc}; while the smallest mean of all target sets, is the minimum score, a_{vc}. The values of a_{vc} and b_{vc} are zero and one, respectively. A linear single dimensional value function was used to determine the value of scores between the maximum and minimum, and the function is shown in Figure A-5. This measure differs from that of *traffic volume* in that a link may have a high capacity but not have a large amount of traffic; perhaps indicating a link that was established with future traffic loads and expandability in mind. The mathematical representation of a target set score is :

$$\frac{\sum_{\forall i \in \text{target set}} \text{voice channel amount of target } i}{\left|\{\text{target set}\}\right|}$$

Additionally,

$$a_{vc} = \min\{\text{target set } \textit{voice channels} \text{ scores}\}$$

$$b_{vc} = \max\{\text{target set } \textit{voice channels} \text{ scores}\}$$

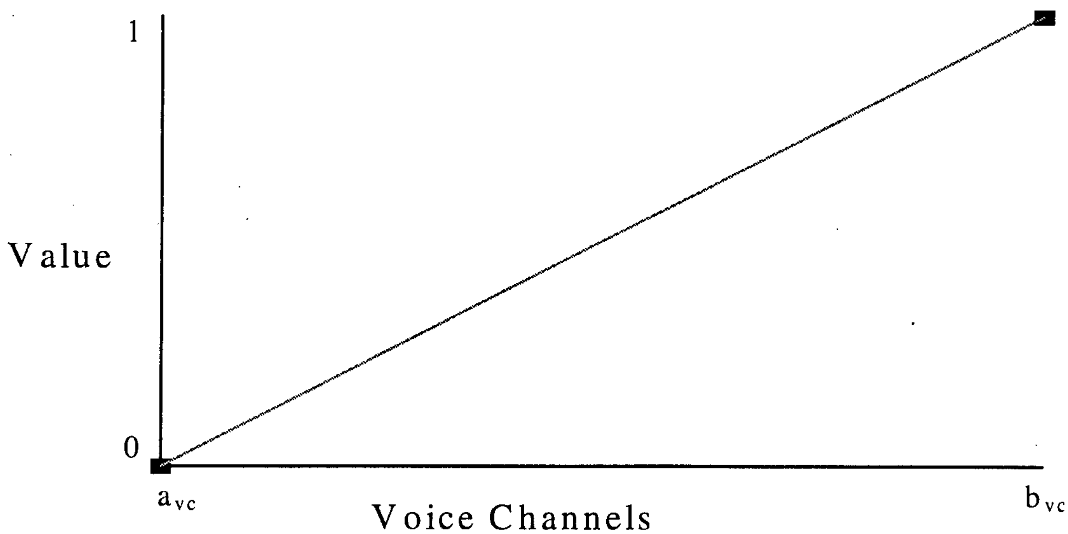

Figure A-5. *Voice Channels* Single Dimensional Value Function

Survivability

This evaluation consideration is composed of two measures: *access* and *mean life*. The objective of this consideration is to choose a target set consisting of targets which are accessible and have a long mean life.

Access: This measure identifies the number of ways that the targets within a set can be accessed. The ideal case occurs if all the targets in the set can be accessed in both a conventional and an unconventional manner. The worst case occurs if there is no way, conventionally or unconventionally, for any of the targets in the set to be accessed. The middle ground between the two cases is when the target can be accessed in either a conventional manner or an unconventional manner, but not both. It is assumed that a target can fall into only one of the above three categories. A target set's score is the mean

of the number of ways that the targets in the set can be accessed. The horizontal axis' scale ranges from a_{access} to b_{access}; where a_{access} corresponds to the lowest target set score, with a value of zero, and b_{access} corresponds to the highest target set score, with a value of one. If a target set contains one or more targets which cannot be accessed, then this set must be carefully scrutinized since the target(s) is(are), for all intents and purposes, screened out of any possible attack. The single dimensional value function is shown in Figure A-6. The target set score is mathematically equivalent to:

$$\frac{\sum_{\forall i \in \text{target set}} \text{number of ways target } i \text{ can be accessed}}{|\{\text{target set}\}|}$$

Additionally,

$$a_{access} = \min\{\text{target set } \mathit{access} \text{ scores}\}$$

$$b_{access} = \max\{\text{target set } \mathit{access} \text{ scores}\}$$

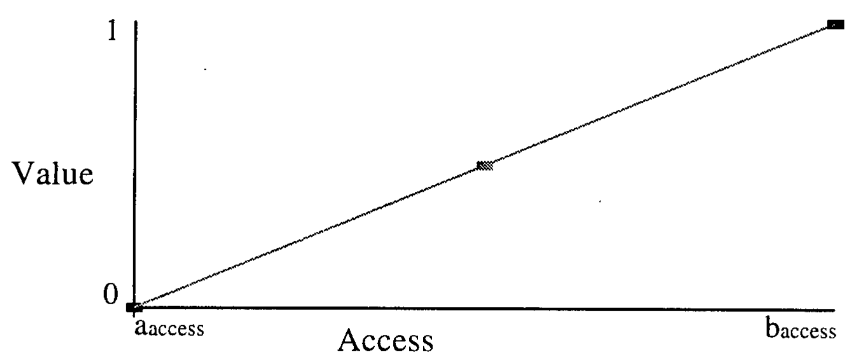

Figure A-6. *Access* Single Dimensional Value Function

Mean Life: This measure determines the value of a target set by the mean of the average service life remaining (ASLR), in years, of each target in the set. If a target set or

part of a target set is attacked, it behooves the attacker to choose a set with a long mean life average. The rationale is that the remaining network components have a shorter ASLR and the likelihood of their replacement, necessitating temporary rerouting or other unwanted situations, increases.

For this study, the ASLR of a node target with multiple, internal components is assumed to be equivalent to the ASLR of the internal component with the shortest ASLR. For example, if a microwave antennae for short haul transmission, a satellite antennae for long haul transmission, and a digital switch were the internal components of the office, then the ASLR of the toll central office node would be the same as the shortest ASLR of the three components. The ASLR of a link target is the minimum of the link media's ASLR and the appropriate component ASLR of the nodes the link joins. For example, suppose a coaxial cable is used to join a central office and a repeater. Next, suppose the coaxial cable has a ASLR of 25 years, the central office component handling the incoming coaxial cable has an ASLR of 5 years, and the repeater has an ASLR of .5 years. The ASLR of the coaxial cable would be .5 years because of the repeater's ASLR.

The measure ranges from a_{ml} to b_{ml}, where a_{ml} corresponds to the lowest target set score, and b_{ml} corresponds to the highest target set score. The values of a_{ml} and b_{ml} are zero and one, respectively. The scale's unit is years. The single dimensional value function is shown in Figure A-7. The mathematical representation of a target set score is:

$$\frac{\sum_{\forall i \in \text{target set}} \text{ASLR of target } i}{\left|\{\text{target set}\}\right|}$$

Additionally,

a_{ml} = min{target set *mean life* scores}

b_{ml} = min{target set *mean life* scores}

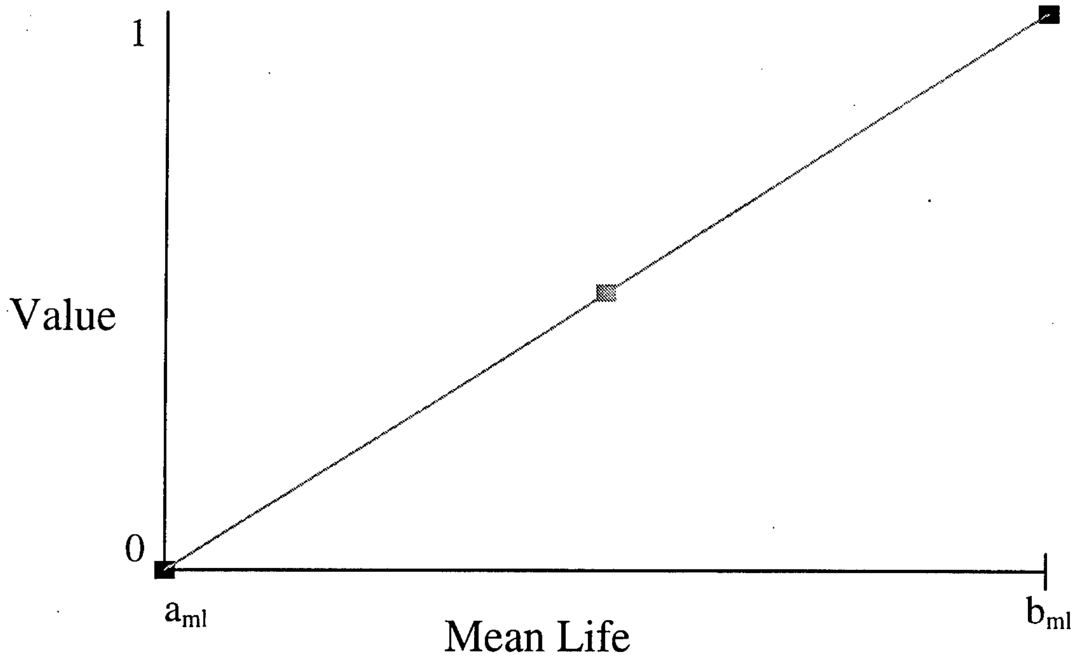

Figure A-7. *Mean Life* Single Dimensional Value Function

Appendix B. First Twenty Cut-sets Generated from Notional Network

CutSet 1	26	28	30	35				
CutSet 2	26	28	30	6				
CutSet 3	26	28	30	33	34			
CutSet 4	26	28	30	33	19			
CutSet 5	26	28	30	33	48	51		
CutSet 6	26	28	30	33	51	17		
CutSet 7	26	28	30	33	51	46	47	49
CutSet 8	26	51	49	18				
CutSet 9	26	51	49	50				
CutSet 10	26	51	49	20				
CutSet 11	26	51	49	52				
CutSet 12	26	51	49	21				
CutSet 13	26	51	49	53				
CutSet 14	51	49	53	1				
CutSet 15	51	49	53	25				
CutSet 16	51	49	53	11				
CutSet 17	51	49	53	41				
CutSet 18	51	49	53	22				
CutSet 19	51	49	53	54				
CutSet 20	51	49	21	1				

Appendix C. First Twenty Target Sets Generated from Notional Network

Target Set 1	Link 1 to 23	Link 2 to 23	Link 3 to 23	Link 6 to 23				
Target Set 2	Link 1 to 23	Link 2 to 23	Link 3 to 23	Node 6				
Target Set 3	Link 1 to 23	Link 2 to 23	Link 3 to 23	Link 6 to 7	Link 6 to 19			
Target Set 4	Link 1 to 23	Link 2 to 23	Link 3 to 23	Link 6 to 7	Node 19			
Target Set 5	Link 1 to 23	Link 2 to 23	Link 3 to 23	Link 6 to 7	Link 17 to 19	Link 19 to 24		
Target Set 6	Link 1 to 23	Link 2 to 23	Link 3 to 23	Link 6 to 7	Link 19 to 24	Node 17		
Target Set 7	Link 1 to 23	Link 2 to 23	Link 3 to 23	Link 6 to 7	Link 19 to 24	Link 16 to 17	Link 17 to 18	Link 17 to 24
Target Set 8	Link 1 to 23	Link 19 to 24	Link 17 to 24	Node 18				
Target Set 9	Link 1 to 23	Link 19 to 24	Link 17 to 24	Link 18 to 20				
Target Set 10	Link 1 to 23	Link 19 to 24	Link 17 to 24	Node 20				
Target Set 11	Link 1 to 23	Link 19 to 24	Link 17 to 24	Link 20 to 21				
Target Set 12	Link 1 to 23	Link 19 to 24	Link 17 to 24	Node 21				
Target Set 13	Link 1 to 23	Link 19 to 24	Link 17 to 24	Link 21 to 24				
Target Set 14	Link 19 to 24	Link 17 to 24	Link 21 to 24	Node 1				
Target Set 15	Link 19 to 24	Link 17 to 24	Link 21 to 24	Link 1 to 11				
Target Set 16	Link 19 to 24	Link 17 to 24	Link 21 to 24	Node 11				
Target Set 17	Link 19 to 24	Link 17 to 24	Link 21 to 24	Link 11 to 22				
Target Set 18	Link 19 to 24	Link 17 to 24	Link 21 to 24	Node 22				
Target Set 19	Link 19 to 24	Link 17 to 24	Link 21 to 24	Link 22 to 24				
Target Set 20	Link 19 to 24	Link 17 to 24	Node 21	Node 1				

Appendix D. Twenty Highest Value Target Sets' Values

Target Set	Cardinality Value	Traffic Volume Value	User Type Value	Voice Channels Value	Access Value	Mean Life Value
7654	.3	.19	.13	.11	.07	.04
5146	.3	.20	.13	.11	.07	.02
7653	.3	.19	.13	.11	.07	.03
7655	.3	.19	.13	.11	.07	.03
5145	.3	.20	.13	.11	.07	.01
5147	.3	.20	.13	.11	.07	.01
8906	.3	.16	.14	.11	.07	.04
6398	.3	.19	.12	.11	.07	.02
8905	.3	.16	.14	.11	.07	.03
8907	.3	.16	.14	.11	.07	.03
6397	.3	.19	.12	.11	.07	.01
6399	.3	.19	.12	.11	.07	.01
6580	.3	.19	.04	.01	.14	.1
4072	.3	.20	.04	.01	.14	.08
7832	.3	.16	.04	.01	.14	.1
5324	.3	.19	.03	.01	.14	.08
2768	.24	.17	.11	.09	.09	.02
2767	.24	.17	.11	.09	.09	.02
2769	.24	.17	.11	.09	.09	.02
7722	.24	.11	.12	.08	.09	.06

Note: For display purposes, the values in Table E-2 are rounded to two digits; therefore, some target set values or value sums may not appear to match those in Chapter 4. This is not the true case, and full precision displays from the model verify this fact.

Appendix E. Intro-help Module

'This module describes how the network disruption modeling tool works
'and gives helpful techniques

'1) Input Data

'Things to keep in mind:

'The graph of interest MUST be placed on the Input1 worksheet

'The graph of interest MUST be in node adjacency format,
'starting at row 1, column 1

'The node adjacency list MUST NOT have blank rows or columns,
'i.e. Do NOT skip rows or columns

'The node adjacency list MUST consist of integer representations of the nodes
'The node adjacency list MUST NOT contain a node labeled 0
'The node adjacency list MUST be in ascending order
'Each node's adjacent nodes MUST be in ascending order
'See the Sample worksheet for an example of proper input for a given graph

'Use the Input button on the toolbar to switch to the Input1 worksheet
'This is the worksheet will the original graph vertex adjacency list
'is entered

'If it is desired to change the weights on the measures, the user must enter them on the
'worksheet Values, B:11 - L:11; otherwise, the default values are those in the thesis
'A Network Disruption Modeling Tool by Captain James A. Leinart, GOR-98M-15.
'If a piecewise linear single dimensional value function is desired then the user
'must enter, in ascending order,the score of the alternative in cells A7:A10,
'C7:C10, and so forth; followed by the corresponding values in cells B7:B10,
'D7:D10, and so forth. These cell changes must be done on the worksheet Values.
'Currently, the model is limited to value functions with three or less pieces.
'If a different shaped exponential single dimensional value function, then the user
'must enter, on the Values worksheet, a different Low, High, Mono, and Rho. Low
'corresponds to the lower bound of the range, High to the upper bound of the range
'Mono indicates whether the curve is increasing or decreasing and Rho indicates the

'shape of the curve (convex, concave, or straight line)
'Additional measures may be added, but involve programming changes in several
'modules. Value function shape changes, weight changes, and adding more measures
'should be accomplished at the graph input step. Toolbar buttons may also have to
'be linked to the appropriate macros, in certain cases.

'2) Graph Transformation

'In order to generate all the target sets, the node adjacency list
'is transformed into an alternate, but equivalent node adjacency list.
'The original node adjacency list MUST be transformed in order for ALL
'target sets to be generated. Transformation of the original node adjacency
'list is accomplished with the X -> Y menu button, and the transformed list
'can be viewed on the Input worksheet. Transformation is valid ONLY IF
'there is a node adjacency list on the Input1 worksheet.

'3) Scoring Data

'Use the Data button to enter target attributes ONLY AFTER transformation of
'the original node adjacency list has been accomplished.
'Target attributes are entered on the Attributes worksheet.

'4) Target Set Generation

'ONLY AFTER the targeting data has been entered in the Attributes worksheet,
'should the target sets be generated. Use the Sets button to generate the
'unconverted (in vertex-only form) target sets. The target sets are listed
'in the CutSets worksheet.

'5) Formatted Target Sets

'ONLY AFTER the Sets button has been used to generate all the unconverted
'target sets, the Form (short for Format) button can be used to list the
'target sets composed of the links and nodes of the original network.

'6) Scores

'Given the scoring data (i.e. target attributes), the target set scores can
'be determined by selecting the Scores menu button on the Analysis toolbar
'This button can be used ONLY AFTER the target sets
'have been generated by using the Sets menu button.

'7) Values

'AFTER the ranges have been determined, the values of each target set can be
'obtained with the use of the values button. Column M contains each target
'set's value and how it scored on each measure

'8) Target Nomination List (TNL)

'AFTER the target set values have been determined, a prioritized candidate target
'list is available by using the TNL menu button
'A choice on the number of ranked targets on the final TNL is offered
'while the draft TNL displays all targets

'NOTE: When dealing with large graphs - nodes >40 and edges > 81
'the user must update the memory allocations in the TargetGen Module
'Specifically, the constant ArrayAmount should be updated to reflect
'the number of nodes of the transformed graph and the array argument
'for EdgeNodes() should be updated to accommodate the number of
'edges in the transformed graph under investigation

'Finally, improper/unpredictable program behavior may result if the
'workbook is not closed entirely, then reopened upon examination of a second
'problem or reexamination of the graph originally of interest. Additionally,
'anomalies in the tool may occur if the steps above are not followed in the
'specified sequence. If review of the tool's analysis results is desired
'open the worksheet corresponding to the results rather than using a toolbar
'button again. Failure to do so may result in improper program operation.

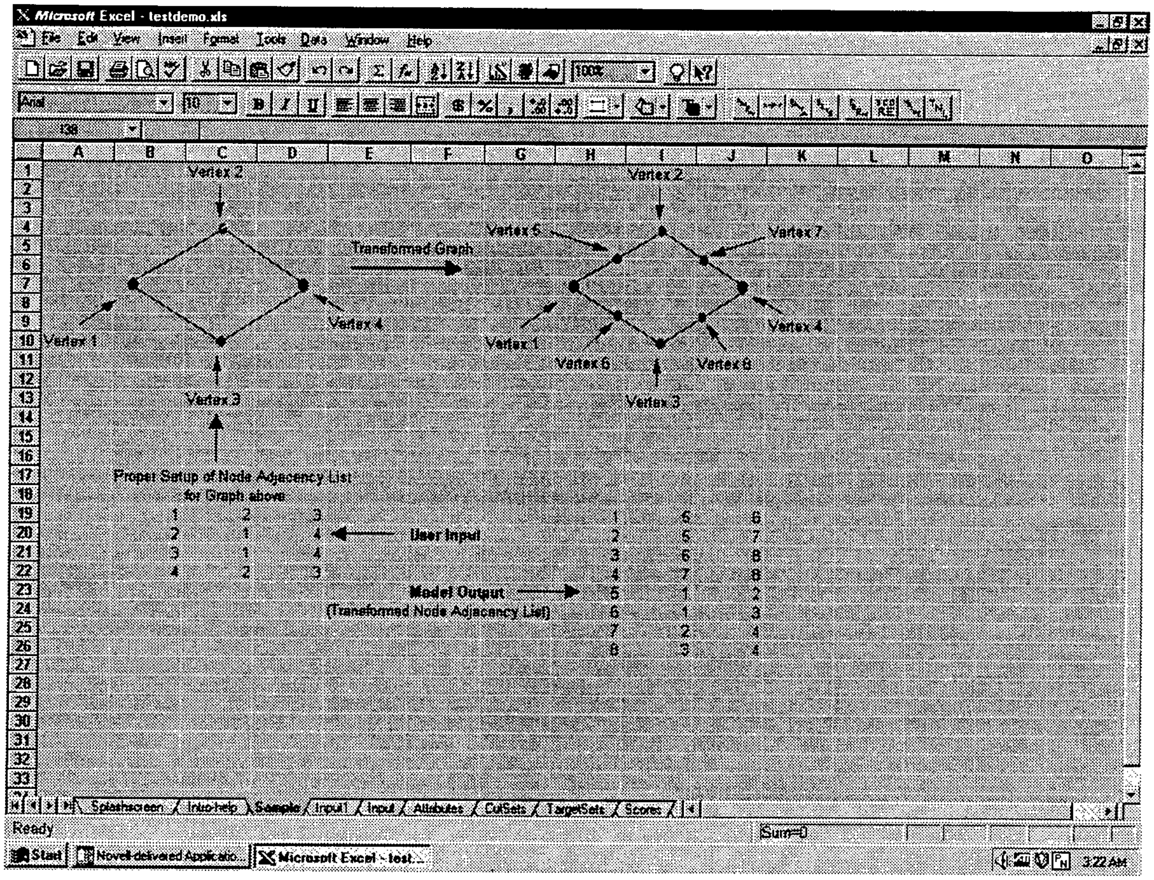

Figure E-1. Worksheet, Sample, and Analysis Toolbar

Appendix F1. InputMod Module

```
'This module activates the sheet, Input1, where the network
'node adjacency list of the network is entered. This module is run when
'the INPUT button on the Analysis tool bar is selected


Sub GraphInput()
   ThisWorkbook.Sheets("Input1").Activate
End Sub
```

Appendix F2. Transformation Module

```
'This module performs the transformation of the entered vertex adjacency list
'so that the list is in the proper format for the cut-set enumeration algorithm.
'This module is run when the X->Y menu button on the Analysis toolbar is chosen.


Option Explicit
Option Base 1


Sub Transform(PreInputSheet)

Dim ListLength As Integer, TempLength As Integer, I As Integer, J As Integer, K As
Integer
Dim L As Integer, Placeholder As Integer


With PreInputSheet

  ListLength = .Range("A:A").CurrentRegion.Rows.Count
  TempLength = ListLength + 1
  J = 2: K = 2: L = 2

  For I = 1 To ListLength

    Sheets("Input").Cells(I, 1) = I

    Do While ((.Cells(I, L) < I) And (.Cells(I, L) <> 0))
      L = L + 1
    Loop

    Do While (.Cells(I, L) <> 0)
      Sheets("Input").Cells(I, L) = TempLength
      Sheets("Input").Cells(TempLength, 1) = TempLength
      Placeholder = .Cells(I, L).Value

      Do While (Sheets("Input").Cells(Placeholder, K) <> 0)
          K = K + 1
      Loop
```

```
        Sheets("Input").Cells(Placeholder, K) = TempLength
        K = 2

        Do While Sheets("Input").Cells(TempLength, K) <> 0
          K = K + 1
        Loop

        Sheets("Input").Cells(TempLength, K) = I
        Sheets("Input").Cells(TempLength, K + 1) = Placeholder
        K = 2
        TempLength = TempLength + 1
        L = L + 1
      Loop

      J = 2: K = 2: L = 2
   Next I

End With

End Sub



Sub Change()

With Sheets("Input")
   .Range("a:iv").ClearContents
End With

   Sheets("Input").Activate
   Call Transform(Sheets("Input1"))

End Sub
```

Appendix F3. ScoreMod Module

```
'This module performs the activation of the Attributes sheet, where the scoring
'data is input by the user. It is selected using the Data menu button on the
'Analysis toolbar.


Option Explicit
Option Base 1


Sub Scoring()

Dim I As Integer, Input1Length As Integer, InputLength As Integer
Dim J As Integer


With Sheets("Attributes")

   .Range("A3:iv16384").ClearContents
   Sheets("Attributes").Activate
   Input1Length = Sheets("Input1").Range("A:A").CurrentRegion.Rows.Count
   InputLength = Sheets("Input").Range("A:A").CurrentRegion.Rows.Count

   For I = 1 To Input1Length
      .Cells(I + 2, 1) = "Node " & I
   Next I

   For J = I To InputLength
      .Cells(J + 2, 1) = "Link " & Sheets("Input").Cells(J, 2) & " to " &
Sheets("Input").Cells(J, 3)
   Next J

   .Cells(3, 2).Activate

End With

End Sub
```

Appendix F4. TargetGen Mod

```
'This module implements the cut-set enumeration algorithm of Patvardhan, Prasad
'and Pyara (1995). A connected components algorithm is needed for step three
'and the connected components algorithm used was derived from a connected
'components algorithm, written in Pascal, in a computer edition of the text
'Foundations of Computer Science by Aho and Ullman.
'This module is run by selecting the Sets menu button on the Analysis tool bar.
'Additionally, this module contains the Auto_Open file that drives the Splashscreen.

Option Explicit
Option Base 1


Type EdgeNode
   node1 As Integer
   node2 As Integer
   next As Integer
End Type


Type TreeNode
   height As Integer
   parent As Integer
End Type


Const ArrayAmount As Integer = 120

Dim Nodes(ArrayAmount) As TreeNode
Dim OuterCount As Integer, Called As Integer, Texit As Integer, Zexit As Integer, Inlist
As Integer
Dim Listflag As Integer, ListCount As Integer
Dim Response1
Dim Msg As String


Sub Auto_Open()

Dim newHour As Integer, newMinute As Integer, newSecond As Integer
Dim waitTime
```

```
    ThisWorkbook.Sheets("Splashscreen").Activate
    newHour = Hour(Now())
    newMinute = Minute(Now())
    newSecond = Second(Now()) + 5
    waitTime = TimeSerial(newHour, newMinute, newSecond)
    Application.Wait waitTime
    ThisWorkbook.Sheets("Intro-help").Activate
    MsgBox ("Please read instructions carefully before proceeding")

End Sub


Sub Cancel_Click()

Dim Config
Dim Title As String

    Msg = "If you cancel now, all data will be lost " & _
    " Are you sure you want to cancel?"
    Config = vbYesNo + vbExclamation + vbDefaultButton2
    Title = "Cancel?"
    Response1 = MsgBox(Msg, Config, Title)

End Sub



'The following procedure is a driver which calls all the appropriate procedures
'in order to take network data from a spreadsheet and output the value cut-sets.


Sub DriverCutSet()

Dim VertexSet(ArrayAmount) As Integer
Dim Tree(ArrayAmount) As Integer
ListCount = 1

With Sheets("CutSets")
    .Range("a:iv").ClearContents
End With

    Sheets("CutSets").Activate
```

```
With Sheets("Input1")
  'This sets the first element of Array VertexSet to the first separated node
  Inlist = .Range("A:A").CurrentRegion.Rows.Count

Input1:
    DialogSheets("InputDialog").EditBoxes("Source").Text = " "
    DialogSheets("InputDialog").Show

    If Response1 = vbYes Then
      Exit Sub
    End If

    If DialogSheets("InputDialog").EditBoxes("Source").Text = " " Then
      GoTo Input1
    End If

    VertexSet(1) = DialogSheets("InputDialog").EditBoxes("Source").Text

    Do While ((ListCount <= Inlist) And (Listflag = 0))

      If .Cells(ListCount, 1) = VertexSet(1) Then
        Listflag = 1
      End If

      ListCount = ListCount + 1
    Loop

    If Listflag = 0 Then
      MsgBox ("This source vertex does not exist - reenter")
      ListCount = 1
      GoTo Input1
    Else
      Listflag = 0
      ListCount = 1
    End If

Input2:
    DialogSheets("Input2Dialog").EditBoxes("Sink").Text = " "
    DialogSheets("Input2Dialog").Show

    If Response1 = vbYes Then
      Exit Sub
    End If
```

```
        If DialogSheets("Input2Dialog").EditBoxes("Sink").Text = " " Then
            GoTo Input2
        End If

        'This sets the first element of the Array Tree to the second separated node
        Tree(1) = DialogSheets("Input2Dialog").EditBoxes("Sink").Text

        If Tree(1) = VertexSet(1) Then
            MsgBox ("Sink and source vertices the same - reenter")
            GoTo Input2
        End If

        Do While ((ListCount <= Inlist) And (Listflag = 0))

            If .Cells(ListCount, 1) = Tree(1) Then
                Listflag = 1
            End If

            ListCount = ListCount + 1
        Loop

        If Listflag = 0 Then
            MsgBox ("This sink vertex does not exist - reenter")
            ListCount = 1
            GoTo Input2
        Else
            Listflag = 0
            ListCount = 1
        End If

End With

    Call Generate(VertexSet(), Tree(), Sheets("Input"))

    Sheets("CutSets").Activate

End Sub


Function FindRoot(passednode)

Dim newp As Integer

newp = passednode
```

```
  Do While (Nodes(newp).parent <> 0)
    newp = Nodes(newp).parent
  Loop
  FindRoot = newp

End Function


Sub MergeTrees(t1, t2)

  If Nodes(t1).height > Nodes(t2).height Then
    Nodes(t2).parent = t1
  Else
    Nodes(t1).parent = t2
  End If

  If Nodes(t1).height = Nodes(t2).height Then
    Nodes(t2).height = Nodes(t2).height + 1
  End If

End Sub



'The following procedure will yield valid cutsets by
'creating unique graphs/subgraphs containing node s

Sub Generate(Vset() As Integer, Tset() As Integer, Opsheet)

Dim A As Integer, Aprime As Integer, B As Integer, Bprime As Integer
Dim D As Integer, E As Integer, F As Integer, G As Integer
Dim H As Integer, I As Integer, J As Integer, K As Integer, L As Integer
Dim M As Integer, Lprime As Integer, N As Integer, O As Integer, P As Integer
Dim Q As Integer, R As Integer, S As Integer, T As Integer, U As Integer
Dim v As Integer, X As Integer
Dim ColumnCount As Integer, RowCount As Integer, Vflag As Integer
Dim Flag1 As Integer, Flag2 As Integer, Flag3 As Integer, Flag4 As Integer
Dim Flag5 As Integer, Flag7 As Integer, FlagU As Integer
Dim Tplus As Integer, Flag8 As Integer, Flag9 As Integer, Flag10 As Integer
Dim Xprime As Integer, Sprime As Integer, Sdprime As Integer
Dim Flagtry As Integer, Limit1 As Integer, Counting As Integer
Dim Flageq As Integer, FlagRow As Integer, CountE As Integer
Dim Vx(ArrayAmount) As Integer
```

```
Dim Z(ArrayAmount) As Integer
Dim Vcut(ArrayAmount) As Integer
Dim Tprime(ArrayAmount) As Integer
Dim Vcminust(ArrayAmount) As Integer
Dim Vtry(ArrayAmount) As Integer
Dim Ttry(ArrayAmount) As Integer
Dim Vsecondtry(ArrayAmount) As Integer
Dim Tsecondtry(ArrayAmount) As Integer
Dim Vtemp(ArrayAmount) As Integer
Dim Edges(198) As EdgeNode
Dim newi As Integer, newk As Integer, r1 As Integer, r2 As Integer
Dim gett As Integer, tempt As Integer
Dim Connect(ArrayAmount) As Integer
Dim Msg
Dim StringOut As String


'Initializations
  A = 1: B = 1: Bprime = 1: D = 1: E = 1: F = 1: G = 1: H = 1
  I = 1: J = 1: K = 2: L = 1: M = 1: Lprime = 1: N = 1: O = 1: P = 1
  S = 1: T = 1: U = 1: ColumnCount = 2: Vflag = 1: RowCount = 1: Q = 1
  R = 1: Aprime = 1: Tplus = 1: X = 1: Xprime = 1: Sprime = 1: Sdprime = 1
  Limit1 = 1: Counting = 1: newk = 1

  For I = 1 To ArrayAmount
    Vtry(I) = Vset(I)
    Ttry(I) = Tset(I)
  Next I

  I = 1

  'Begin Step 1 of the algorithm
  With Sheets("Output")
    .Range("a:iv").ClearContents

    Do While (Vtry(I) <> 0)

      For RowCount = 1 To Opsheet.Range("A:A").CurrentRegion.Rows.Count

        If Opsheet.Cells(RowCount, 1) = Vtry(I) Then
          Exit For 'Found a vertex in Vset
        End If

      Next RowCount
```

```
'Now that a vertex in Vset has been found, traverse its adjacency list
Do While (Opsheet.Cells(RowCount, ColumnCount) <> 0)

   Do While (Vtry(Aprime) <> 0)

      If Opsheet.Cells(RowCount, ColumnCount) = Vtry(Aprime) Then
         Vflag = 0
         Exit Do 'A vertex does not belong in Vx
      End If

      Aprime = Aprime + 1
   Loop

   'A vertex belongs in Vx but make sure it's not already in there
   If Vflag = 1 Then

  Do While ((Vx(Bprime) <> 0) And (Bprime <= ArrayAmount) And (Vflag = 1))

         If Opsheet.Cells(RowCount, ColumnCount) <> Vx(Bprime) Then
            Bprime = Bprime + 1
         Else
            Vflag = 0
         End If

      Loop 'Bprime

      Bprime = 1

         If Vflag = 1 Then
         Vx(B) = Opsheet.Cells(RowCount, ColumnCount)
         B = B + 1
         End If

   End If

   Aprime = 1
   Vflag = 1
   ColumnCount = ColumnCount + 1
Loop 'ColumnCount

I = I + 1
ColumnCount = 2
```

```
 Loop 'I
'End Step 1 of the algorithm


'Begin Step 2 - If t is an element of Vx goto OVER
Do While ((Vx(D) <> 0) And (D <= ArrayAmount))

   If Vx(D) = DialogSheets("Input2Dialog").EditBoxes("Sink").Text Then
      Texit = Texit + 1
      GoTo OVER
   Else
      D = D + 1
   End If

Loop
'End Step 2 of the algorithm



'Begin Step 3 of the algorithm
'See which components are connected to t but aren't in either Vset or Vx
RowCount = 1
ColumnCount = 2

'Find V-(Vtry U Vx)
For RowCount = 1 To Opsheet.Range("A:A").CurrentRegion.Rows.Count

   Do While ((Flag1 = 0) And (Vtry(E) <> 0) And (E <= ArrayAmount))

      If Opsheet.Cells(RowCount, 1) = Vtry(E) Then
         Flag1 = 1
      Else
         E = E + 1
      End If
   Loop

   If Flag1 = 0 Then
      E = 1

      Do While ((Flag1 = 0) And (Vx(E) <> 0) And (E <= ArrayAmount))

         If Opsheet.Cells(RowCount, 1) = Vx(E) Then
            Flag1 = 1
         Else
```

```
            E = E + 1
         End If

      Loop

   End If

   If Flag1 = 0 Then
      Vtemp(F) = Opsheet.Cells(RowCount, 1)
      F = F + 1
   End If

   Flag1 = 0
   E = 1
Next RowCount


'Found V-(Vtry U Vx)
'Obtain edges of Vtemp (subgraph of t) and assign them to data structure
I = 1
Do While ((Vtemp(G) <> 0) And (G <= ArrayAmount))

   Do While (Vtemp(G) <> Opsheet.Cells(I, 1))

      I = I + 1
   Loop

   J = G + 1

   Do While ((Vtemp(J) <> 0) And (J <= ArrayAmount))

      Do While ((Opsheet.Cells(I, K) <> 0) And (FlagRow = 0))

         If Vtemp(J) = Opsheet.Cells(I, K) Then
            Edges(E).node1 = Opsheet.Cells(I, 1)
            Edges(E).node2 = Opsheet.Cells(I, K)
            FlagRow = 1
            CountE = E - 1

            If E <> 1 Then
               Edges(CountE).next = E
               Edges(E).next = 0
            Else
               Edges(E).next = 0
```

```
            End If

          E = E + 1
          End If

        K = K + 1
        Loop

    FlagRow = 0
    K = 2
    J = J + 1
    Loop

G = G + 1
I = 1
Loop

E = 1


'Call procedure BuildConnectedComponents
For newi = 1 To ArrayAmount
   Nodes(newi).parent = 0
   Nodes(newi).height = 0
Next newi

Do While (Edges(newk).node1 <> 0)

   r1 = FindRoot(Edges(newk).node1)
   r2 = FindRoot(Edges(newk).node2)

   If r1 <> r2 Then
      Call MergeTrees(r1, r2)
   End If

   newk = Edges(newk).next

   If newk = 0 Then
      Exit Do
   End If

Loop

J = 1: I = 1
```

```
gett = FindRoot(DialogSheets("Input2Dialog").EditBoxes("Sink").Text)

Do While (Opsheet.Cells(I, 1) <> 0)
   tempt = FindRoot(Opsheet.Cells(I, 1))

   If tempt = gett Then
      Connect(J) = Opsheet.Cells(I, 1)
      J = J + 1
   End If

I = I + 1
Loop

.Range("a:iv").ClearContents
'End step 3 of the algorithm



'This loop does step 4 of the algorithm
'Checks to see if there is a path from Vx to Vt
RowCount = 1: ColumnCount = 2: J = 1: K = 1

Do While ((Vx(H) <> 0) And (H <= ArrayAmount))

   'Find the row of the vertex in Vx
   Do While (Vx(H) <> Opsheet.Cells(RowCount, 1))
      RowCount = RowCount + 1
   Loop

   Do While ((Opsheet.Cells(RowCount, ColumnCount) <> 0) And (Flag3 = 0))

      Do While ((Flag2 = 0) And (J <= ArrayAmount))

         If Opsheet.Cells(RowCount, ColumnCount) = Connect(J) Then
            Flag2 = 1: Flag3 = 1
         Else
            J = J + 1
         End If

      Loop

      If Flag2 = 0 Then
         ColumnCount = ColumnCount + 1
         J = 1
```

```
        End If

      Loop

    If Flag3 = 0 Then
      Z(K) = Vx(H)
      K = K + 1
    End If

    Flag2 = 0: Flag3 = 0
    H = H + 1
    J = 1
    RowCount = 1: ColumnCount = 2
    Loop

    K = 1

    End With
    'End step 4 of the algorithm


    'Begin step 5 of the algorithm
    With Sheets("Output")

    Do While ((Z(L) <> 0) And (L <= ArrayAmount) And (Flag4 = 0))

      Do While ((Ttry(M) <> 0) And (M <= ArrayAmount) And (Flag4 = 0))

        If Z(L) <> Ttry(M) Then
          M = M + 1
        Else
          Flag4 = 1
        End If

      Loop
      M = 1
      L = L + 1

    Loop

    If Flag4 = 1 Then
      Zexit = Zexit + 1
      GoTo OVER
    End If
```

```
End With
'End Step 5 of the algorithm



'Begin Step 6 of the algorithm
L = 1: M = 1
With Sheets("Output")

Do While (Vtry(L) <> 0)
  L = L + 1
Loop

Do While ((Z(M) <> 0) And (M <= ArrayAmount))

  Do While (Vtry(Lprime) <> 0 And (Lprime <= ArrayAmount) And (FlagU = 0))

    If Z(M) <> Vtry(Lprime) Then
      Lprime = Lprime + 1
    Else
      FlagU = 1
    End If

  Loop

  If FlagU = 0 Then
    Vtry(L) = Z(M)
    L = L + 1
  End If

  Lprime = 1
  M = M + 1
  FlagU = 0
Loop

L = 1

End With
'End Step 6 of the algorithm



'Begin step 7 and 8 of the algorithm
```

```
With Sheets("Output")

 Do While ((Vx(N) <> 0) And (N <= ArrayAmount))

   Do While ((Z(O) <> 0) And (O <= ArrayAmount) And (Flag5 = 0))

     If Vx(N) <> Z(O) Then
       O = O + 1
     Else
       Flag5 = 1
     End If

   Loop

   If Flag5 = 0 Then
     Vcut(P) = Vx(N)
     P = P + 1
   End If

   O = 1
   N = N + 1
   Flag5 = 0
 Loop

 P = 1

 If Vcut(P) <> 0 Then
   OuterCount = OuterCount + 1
 End If

 Do While ((Vcut(P) <> 0) And (P <= ArrayAmount))

   .Cells(21, P + 1) = Vcut(P)
   With Sheets("CutSets")
     .Cells(OuterCount, P + 1) = Vcut(P)
   End With
   P = P + 1
 Loop

 With Sheets("CutSets")
 .Cells(OuterCount, 1) = "CutSet " & OuterCount
 End With
```

```
End With
'End Step 7 and 8 of the algorithm



'Begin Step 10-13 of the algorithm
N = 1: O = 1: P = 1 ': Test = 7

With Sheets("Output")

   'Step 10 begins with determining Vcminust = Vc - T
      Do While ((Vcut(N) <> 0) And (N <= ArrayAmount))

         Do While ((Ttry(O) <> 0) And (O <= ArrayAmount) And (Flag7 = 0))

            If Vcut(N) <> Ttry(O) Then
               O = O + 1
            Else
               Flag7 = 1
            End If

         Loop

         If Flag7 = 0 Then
            Vcminust(P) = Vcut(N)
            P = P + 1
         End If

         O = 1
         N = N + 1
         Flag7 = 0
      Loop



      'Start step 11 of the algorithm if Vc - T is not the empty set
      Do While (Vcminust(1) <> 0)
         R = 1: S = 1: T = 1: X = 1: U = 1: v = 0: Flag8 = 0: Flag9 = 0: Flag10 = 0
         Tplus = 1: Xprime = 1: Sprime = 1: Sdprime = 1

         Do While (Vcminust(R + 1) <> 0)
            R = R + 1
         Loop
         'Select vertex v
```

```
v = Vcminust(R)
Vcminust(R) = 0

Do While (Vtry(S) <> 0) 'And (S <= ArrayAmount) And (Flag8 = 0))
   S = S + 1
Loop

For Sprime = 1 To ArrayAmount
   Vsecondtry(Sprime) = Vtry(Sprime)
   Tsecondtry(Sprime) = Ttry(Sprime)
Next Sprime

   Vsecondtry(S) = v

'Add Tset with Tprime
S = 1

Do While (Tprime(U) <> 0) 'Not necessary to UNION

   Do While ((Tsecondtry(Tplus) <> 0) And (Tplus <= ArrayAmount) And
(Flag9 = 0))

      If Tprime(U) <> Tsecondtry(Tplus) Then
         Tplus = Tplus + 1
      Else
         Flag9 = 1
      End If

   Loop

   If Flag9 = 0 Then
      Tsecondtry(Tplus) = Tprime(U)
   End If

   Tplus = 1
   U = U + 1
   Flag9 = 0
Loop


'Step 12 - Recursion
Called = Called + 1
Call Generate(Vsecondtry(), Tsecondtry(), Sheets("Input"))
I = 1
```

```
            'End Step 12

            'Step 13
            U = 1
            Do While ((Tprime(U) <> 0) And (U <= ArrayAmount) And (Flag10 = 0))

              If Tprime(U) <> v Then
                U = U + 1
              Else
                Flag10 = 1
              End If

            Loop

            If Flag10 = 0 Then
              Tprime(U) = v
            End If

        Loop

    End With

OVER:
End Sub
```

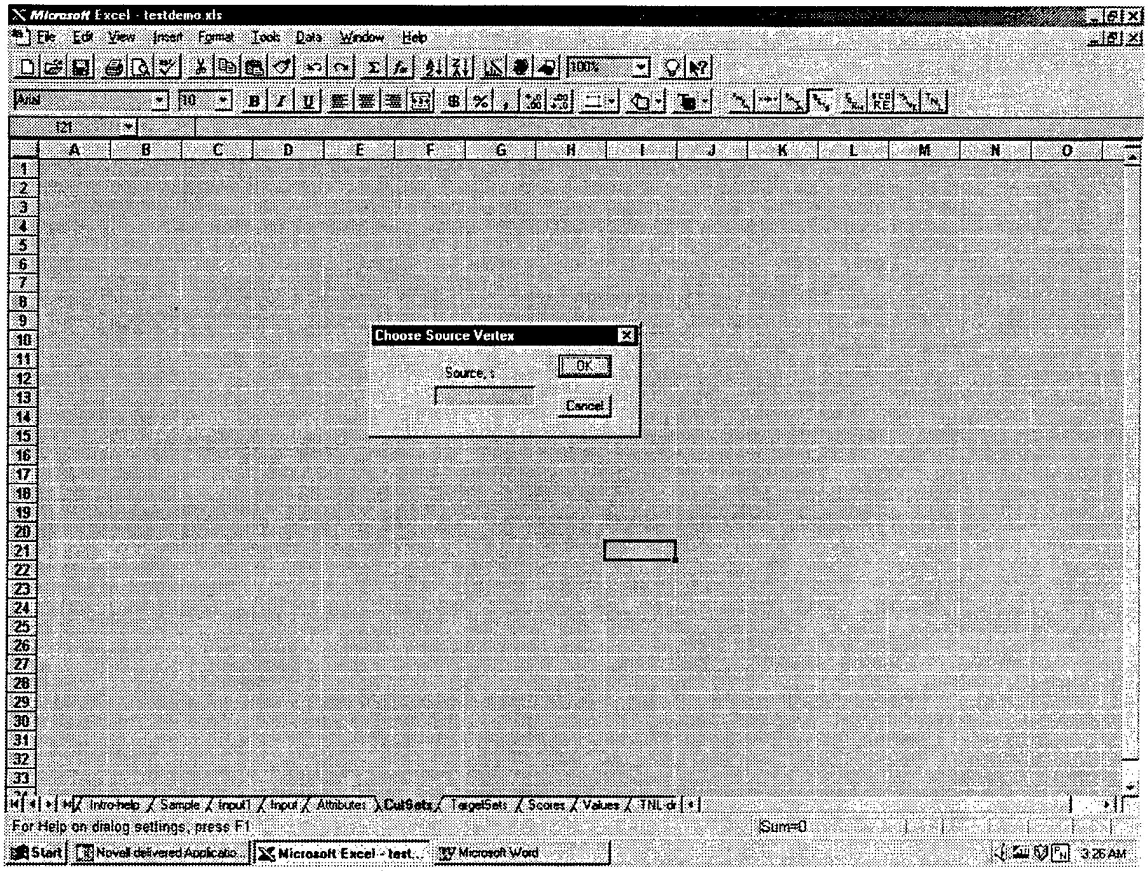

Figure F4-1. Dialog Box for Source Vertex in CutSets Sheet

Appendix F5. FormatMod

```
'This module performs the converts the cut-set enumeration algorithm output
'into the target set format of network nodes and links.


Option Explicit
Option Base 1


Sub Formatting()

Dim I As Integer, J As Integer, setlength As Integer, temp As Integer


   setlength = Sheets("CutSets").Range("a:iv").CurrentRegion.Rows.Count
   J = 2
   Sheets("TargetSets").Range("a:iv").ClearContents
   Sheets("TargetSets").Activate

   With Sheets("TargetSets")

   For I = 1 To setlength
      .Cells(I, 1) = "Target Set " & I

      Do While (Sheets("CutSets").Cells(I, J) <> 0)
         temp = Sheets("CutSets").Cells(I, J).Value
         .Cells(I, J) = Sheets("Attributes").Cells(temp + 2, 1)
         J = J + 1
      Loop

      J = 2
   Next I

   End With

End Sub
```

Appendix F6. RangesMod

```
'This module calculates the ranges of the single dimensional value functions and
'determines the score of each target set


Option Explicit
Option Base 1


Dim NumCutsets As Integer


Sub CardinalityRange(CutsetCount As Integer)

Dim I As Integer, J As Integer, CardinalityArray() As Integer

ReDim CardinalityArray(1 To CutsetCount)

J = 2

  For I = 1 To CutsetCount
    Do While (Sheets("CutSets").Cells(I, J) <> 0)
        CardinalityArray(I) = CardinalityArray(I) + 1
        J = J + 1
    Loop
    With Sheets("Scores")
      .Cells(I + 1, 1) = "Target Set" & I
      .Cells(I + 1, 2) = CardinalityArray(I)
    End With
    J = 2
  Next I
End Sub


Sub TrafficRange(CutsetCount As Integer)

Dim I As Integer, J As Integer, TrafficArray() As Single
Dim traffic_counter As Single, temp As Integer, traffic_average As Single

ReDim TrafficArray(1 To CutsetCount)
```

```
J = 2

   For I = 1 To CutsetCount
      Do While (Sheets("CutSets").Cells(I, J) <> 0)
         temp = Sheets("CutSets").Cells(I, J).Value
         traffic_counter = traffic_counter + Sheets("Attributes").Cells(temp + 2, 2)
         J = J + 1
      Loop
      traffic_average = traffic_counter / (J - 2)
      TrafficArray(I) = traffic_average
      With Sheets("Scores")
         .Cells(I + 1, 3) = Application.Round(TrafficArray(I), 2)
      End With
      traffic_counter = 0
      J = 2
   Next I

End Sub


Sub UserTypeRange(CutsetCount As Integer)

Dim I As Integer, J As Integer, UserTypeArray() As Single
Dim usertype_counter As Single, temp As Integer, usertype_average As Single

ReDim UserTypeArray(1 To CutsetCount)

J = 2

   For I = 1 To CutsetCount
      Do While (Sheets("CutSets").Cells(I, J) <> 0)
         temp = Sheets("CutSets").Cells(I, J).Value
         usertype_counter = usertype_counter + Sheets("Attributes").Cells(temp + 2, 3)
         J = J + 1
      Loop
      usertype_average = usertype_counter / (J - 2)
      UserTypeArray(I) = usertype_average
      With Sheets("Scores")
         .Cells(I + 1, 4) = Application.Round(UserTypeArray(I), 2)
      End With
      usertype_counter = 0
      J = 2
   Next I
```

```
End Sub


Sub VoiceChannelRange(CutsetCount As Integer)

Dim I As Integer, J As Integer, VoiceChannelArray() As Single
Dim voicechannel_counter As Single, temp As Integer, voicechannel_average As Single

ReDim VoiceChannelArray(1 To CutsetCount)

J = 2

  For I = 1 To CutsetCount
     Do While (Sheets("CutSets").Cells(I, J) <> 0)
        temp = Sheets("CutSets").Cells(I, J).Value
        voicechannel_counter = voicechannel_counter + Sheets("Attributes").Cells(temp
+ 2, 4)
        J = J + 1
     Loop
     voicechannel_average = voicechannel_counter / (J - 2)
     VoiceChannelArray(I) = voicechannel_average
     With Sheets("Scores")
        .Cells(I + 1, 5) = Application.Round(VoiceChannelArray(I), 2)
     End With
     voicechannel_counter = 0
     J = 2
  Next I

End Sub


Sub AccessRange(CutsetCount As Integer)

Dim I As Integer, J As Integer, AccessArray() As Single
Dim access_counter As Single, temp As Integer, access_average As Single

ReDim AccessArray(1 To CutsetCount)

J = 2

  For I = 1 To CutsetCount
     Do While (Sheets("CutSets").Cells(I, J) <> 0)
        temp = Sheets("CutSets").Cells(I, J).Value
```

```
        access_counter = access_counter + Sheets("Attributes").Cells(temp + 2, 5)
        J = J + 1
      Loop
      access_average = access_counter / (J - 2)
      AccessArray(I) = access_average
      With Sheets("Scores")
        .Cells(I + 1, 6) = Application.Round(AccessArray(I), 2)
      End With
      access_counter = 0
      J = 2
    Next I

End Sub


Sub MeanLifeRange(CutsetCount As Integer)

Dim I As Integer, J As Integer, MeanLifeArray() As Single
Dim meanlife_counter As Single, temp As Integer, meanlife_average As Single

ReDim MeanLifeArray(1 To CutsetCount)

J = 2

    For I = 1 To CutsetCount
      Do While (Sheets("CutSets").Cells(I, J) <> 0)
        temp = Sheets("CutSets").Cells(I, J).Value
        meanlife_counter = meanlife_counter + Sheets("Attributes").Cells(temp + 2, 6)
        J = J + 1
      Loop
      meanlife_average = meanlife_counter / (J - 2)
      MeanLifeArray(I) = meanlife_average
      With Sheets("Scores")
        .Cells(I + 1, 7) = Application.Round(MeanLifeArray(I), 2)
      End With
      meanlife_counter = 0
      J = 2
    Next I

End Sub


Sub RangesMaxMin(CutsetCount As Integer)
```

```
Dim Min1 As Single, Max1 As Single

    Min1 = Application.Min(Sheets("Scores").Range("B:B"))
    Sheets("Values").Cells(7, 2) = Min1
    Max1 = Application.Max(Sheets("Scores").Range("B:B"))
    Sheets("Values").Cells(8, 2) = Max1

    Min1 = Application.Min(Sheets("Scores").Range("C:C"))
    Sheets("Values").Cells(7, 4) = Application.Round(Min1, 2)
    Max1 = Application.Max(Sheets("Scores").Range("C:C"))
    Sheets("Values").Cells(8, 4) = Application.Round(Max1, 2)

    Min1 = Application.Min(Sheets("Scores").Range("D:D"))
    Sheets("Values").Cells(7, 6) = Application.Round(Min1, 2)
    Max1 = Application.Max(Sheets("Scores").Range("D:D"))
    Sheets("Values").Cells(8, 6) = Application.Round(Max1, 2)

    Min1 = Application.Min(Sheets("Scores").Range("E:E"))
    Sheets("Values").Cells(7, 8) = Application.Round(Min1, 2)
    Max1 = Application.Max(Sheets("Scores").Range("E:E"))
    Sheets("Values").Cells(8, 8) = Application.Round(Max1, 2)

    Min1 = Application.Min(Sheets("Scores").Range("F:F"))
    Sheets("Values").Cells(7, 10) = Application.Round(Min1, 2)
    Max1 = Application.Max(Sheets("Scores").Range("F:F"))
    Sheets("Values").Cells(8, 10) = Application.Round(Max1, 2)

    Min1 = Application.Min(Sheets("Scores").Range("G:G"))
    Sheets("Values").Cells(7, 12) = Application.Round(Min1, 2)
    Max1 = Application.Max(Sheets("Scores").Range("G:G"))
    Sheets("Values").Cells(8, 12) = Application.Round(Max1, 2)

End Sub

Sub RangeDeterminations()
  NumCutsets = Sheets("CutSets").Range("a:iv").CurrentRegion.Rows.Count
  Sheets("Scores").Range("a2:iv16384").ClearContents
  Sheets("Scores").Activate
  Call CardinalityRange(NumCutsets)
  Call TrafficRange(NumCutsets)
  Call UserTypeRange(NumCutsets)
  Call VoiceChannelRange(NumCutsets)
  Call AccessRange(NumCutsets)
  Call MeanLifeRange(NumCutsets)
```

```
    Call RangesMaxMin(NumCutsets)

End Sub
```

Appendix F7. LevelsMod

```
'This module calculates the value of each target set. Values for each target set's measures
'are not rounded, but each target set's final overall value is rounded to decimal places.


Option Explicit
Option Base 1



Sub Leveling()

Const NumMeasures As Integer = 6
Dim Numscores As Integer, I As Integer, J As Integer, K As Integer
Dim temp As Integer, L As Integer, placer As Integer
Dim Msg


Numscores = Sheets("CutSets").Range("a:iv").CurrentRegion.Rows.Count

Sheets("Values").Activate

With Sheets("Values")
  Range("a14:iv16384").ClearContents

For I = 1 To Numscores
  .Cells(I + 13, 1) = "Target Set " & I

  For J = 1 To NumMeasures
     .Cells(I + 13, (2 * J)) = Application.Cells(11, 2 * J).Value *
ValueE(Sheets("Scores").Cells(I + 1, J + 1).Value, .Cells(7, 2 * J).Value, .Cells(8, 2 *
J).Value, .Cells(9, 2 * J).Value, .Cells(10, 2 * J).Value)
     .Cells(I + 13, 13) = .Cells(I + 13, 13) + .Cells(I + 13, 2 * J)
  Next J

  .Cells(I + 13, 13) = Application.Round(.Cells(I + 13, 13), 2)
Next I

End With

End Sub
```

Appendix F8. Module1 [Kirkwood, 1997]

```
Function ValuePL(X, Xi, Vi)
 I = 2
 Do While X > Xi(I)
  I = I + 1
 Loop
 ValuePL = Vi(I - 1) _
  + (Vi(I) - Vi(I - 1)) * (X - Xi(I - 1)) / (Xi(I) - Xi(I - 1))
End Function

Function ValueE(X, Low, High, Monotonicity, Rho)
 Select Case UCase(Monotonicity)
  Case "INCREASING"
   Difference = X - Low
  Case "DECREASING"
   Difference = High - X
 End Select
 If UCase(Rho) = "INFINITY" Then
  If (High - Low) = 0 Then
    ValueE = 0.5
  Else
  ValueE = Difference / (High - Low)
  End If
 Else
  ValueE = (1 - Exp(-Difference / Rho)) / (1 - Exp(-(High - Low) / Rho))
 End If
End Function
```

Appendix F9. TNLMod

```
'This module ranks the target sets on the value sheets by their values and also
'gets input from the users concerning how many target sets to include in the final
'list of candidate target sets


Dim Response1

Sub Norank()

Dim Config
Dim Title As String

   Msg = "If you cancel now, all data will be lost " & _
   " Are you sure you want to cancel?"
   Config = vbYesNo + vbExclamation + vbDefaultButton2
   Title = "Cancel?"
   Response1 = MsgBox(Msg, Config, Title)

End Sub



Sub Nomination()

Const FirstRow As Integer = 14
Dim Numscores As Integer, I As Integer, RowPlace As Integer
Dim A, numrank As Integer
Dim Msg

Numscores = Sheets("CutSets").Range("a:iv").CurrentRegion.Rows.Count
RowPlace = FirstRow
Sheets("TNL-draft").Range("a:iv").ClearContents
Sheets("TNL-final").Range("a:iv").ClearContents
Sheets("TNL-draft").Activate

For I = 1 To Numscores
   Sheets("TNL-draft").Cells(I, 1) = Sheets("Values").Cells(RowPlace, 1)
   Sheets("TNL-draft").Cells(I, 2) = Sheets("Values").Cells(RowPlace, 13)
   RowPlace = RowPlace + 1
Next I
```

```
DialogSheets("Ranking").EditBoxes("EditRank").Text = ""

dialrank:

    DialogSheets("Ranking").Show
    If Response1 = vbYes Then
        Exit Sub
    Else
        A = DialogSheets("Ranking").EditBoxes("EditRank").Text
    End If

    If A = "" Then
        GoTo dialrank
    End If

    If A > Numscores Then
        Msg = "There are not that many existing target sets"
        MsgBox Msg
        GoTo dialrank
    End If
    Worksheets("TNL-draft").Range("a:iv").CurrentRegion.Sort _
    key1:=Worksheets("TNL-draft").Columns("B"), order1:=xlDescending

    Sheets("TNL-final").Activate
    For I = 1 To A
        Sheets("TNL-final").Cells(I, 1) = Sheets("TNL-draft").Cells(I, 1)
        Sheets("TNL-final").Cells(I, 2) = Sheets("TNL-draft").Cells(I, 2)
    Next I

End Sub
```

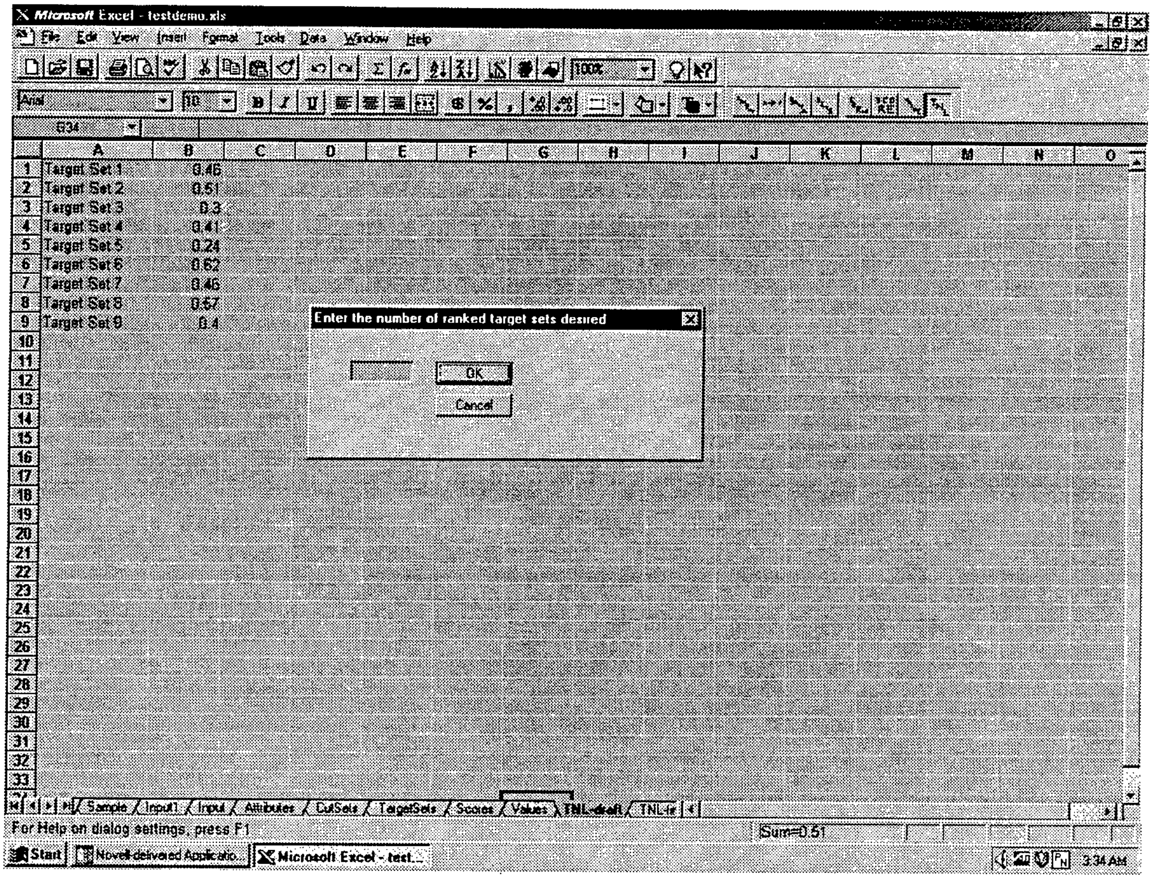

Figure F9-1. Dialog Box for Number of Ranked Target Sets

Vita

Captain Andy Leinart was born November 19, 1969 in Newport News, Virginia. He graduated from Hampton Christian High School in 1987, and attended the University of Virginia in Charlottesville, Virginia. He graduated from the University of Virginia with a Bachelor of Science degree in Applied Mathematics and received his reserve commission in May 1991. He was then assigned to Grand Forks AFB as a Minuteman III ICBM Launch Officer. He served in Grand Forks until July 1996 and then entered the Air Force Institute of Technology in August 1996. Captain Leinart's follow-on assignment was to the Information Warfare Center at Kelly Air Force Base in San Antonio, Texas.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	March, 1998	Masters Thesis

4. TITLE AND SUBTITLE
A Network Disruption Modeling Tool

5. FUNDING NUMBERS

6. AUTHOR(S)
Captain James A. Leinart

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
2750 P. Street
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GOR/ENS/98M-15

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Air Intelligence Center
4180 Watson Way
WPAFB OH 45433

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Given that network disruption has been identified as a military objective and C2-attack has been identified as the mechanism to accomplish this objective, a target set must be acquired and priorities assigned to each target set. In order to move beyond subjective target prioritization, development and implementation of a methodology and model, that quantitatively measures the value of each target in achieving the objective, is needed in prioritizing targets.

This thesis effort uses a vertex cut-set algorithm on a transformation of the graph representing a notional network of interest. The cut-sets generated represent potential target sets. Value-focused thinking and decision analysis techniques are used to rank the target sets according to decision maker preferences and the overall objective of achieving network disruption. The ranked list of potential targets can be narrowed down to an amount palatable to the decision maker, C2W planner, or some other user, so that further analysis may be conducted.

The above methodology is incorporated in a Visual Basic/Excel Spreadsheet environment and allows for user-friendly, yet powerful analysis.

14. SUBJECT TERMS
Network Disruption; Command and Control Warfare; Telecommunications; Value-Focused Thinking; Cut-sets; Targeting

15. NUMBER OF PAGES
123

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL